PB 206 143

COMPUTER MAPPING OF COAL RESERVES BY SULFUR LEVEL: GENERAL DOCUMENTATION PART II

Harvard School of Design
Cambridge, Massachusetts

April 1971

DISTRIBUTED BY:

NTIS
National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

APTD 0729

PB 206 143

COMPUTER MAPPING OF COAL RESERVES BY SULFUR LEVEL: GENERAL DOCUMENTATION

PART II

LABORATORY FOR COMPUTER GRAPHICS
AND SPATIAL ANALYSIS

Contract No. CPA 70-16
Air Pollution Control Office
Environmental Protection Agency

Harvard Graduate School of Design
Cambridge, Massachusetts
April 1971

PRICES SUBJECT TO CHANGE

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
Springfield, Va. 22151

283

COMPUTER MAPPING OF COAL RESERVES BY SULFUR LEVEL

Principal Investigators

John C. Goodrich
Howard T. Fisher*

Study Area Investigators

David Sheehan - Phases I and II (Appalachian Coal Region)
Timothy Murray - Phase III (Allegany and Garrett Counties)

Technical Guidance on Geology

Michael Woldenberg

Research Assistants

Whitmore John
Gail Howrigan

Computer Programming

Nancy Peyton

Secretarial and Administration

Lois Kramer
Tina McGeary

Report Preparation

Albert Davis
Laurence Yont

* thru June 30, 1970

PREFACE

The Air Pollution Control Office (APCO) of the Environmental Protection Agency is interested in ways of displaying the quantity of coal available at various sulfur levels. It was felt that computer mapping of the coal data might provide a relatively quick and inexpensive tool.

Data was provided by the Bureau of Mines for study areas in the Appalachian Coal Region and, in particular, in a two county area in Maryland. Values for sulfur content, bed thickness, and quantity of coal were manipulated and mapped using several computer mapping programs available at the Laboratory for Computer Graphics and Spatial Analysis at the Harvard Graduate School of Design. Although the work was carried out for these study areas, the main emphasis of the project was on providing a tool with general applicability to coal data anywhere in the United States.

The problems of obtaining sufficient reliable data for mapping cannot be emphasized strongly enough. At first glance it would appear that there is a great deal of coal data available. However, when the data are broken down by type, coal seam, and location, there is actually very little useable information available at the present time for all the types of data needed for a study such as this.

It is felt, and has been stressed throughout, that more meaningful results could have been attained if there had been a more equal distribution of data points and a more equal number of analyses taken at these points. All of us were aware that this might be a problem at the outset of the project and our fears have been substantiated. We hope that pointing this out is the kind of constructive criticism that may prove most helpful in overcoming deficiencies in data acquisition so that more meaningful results may be attained in future work.

Of significant importance, however, is the potential applicability of each of the types of maps produced and their usefulness to APCO as adequate data are made available.

The Study Area Report is divided into three main sections, entitled:

- I BACKGROUND
- II STUDY AREA FINDINGS
- III APPENDICES

The first section covers the problem definition, computer program used, study area selection and data acquisition, and the graphic techniques used. The second section discusses the study areas and data used, and the computer maps produced in some detail. The final section contains the step by step description of the mapping procedures used for the study areas, the references, and the technical aspects of program development.

The Study Area Report does not contain any speculative information as to possible future uses of the techniques nor does it include all the computer program manuals and specifications necessary for application of the technique to other coal data and study areas. These manuals and related information are contained in this volume, entitled General Documentation.

Volume Two: General Documentation is divided into two parts. Part I covers the computer mapping techniques and Part II comprises the computer program manuals.

Part I contains seven sections: introduction, computer mapping programs, SYMAP mapping techniques, GRID and SYMVU techniques, discussion of study area graphics, additional data considerations, and data handling routines. This includes all of the general material necessary to understand the techniques which were developed and tested for the study area, and the means for applying them to other study areas. It should be read in conjunction with the Study Area Report and Part II.

Part II contains eight sections: introduction to SYMAP, preparation of packages, map package electives, user subroutine and advanced uses, programming and systems guide, the SYMVU program, the GRID program, and program modifications. This includes all of the technical material necessary to prepare, interpret, and use the three programs, as well as the program modifications made during the project.

We would like to acknowledge the invaluable support and advice of Russell Flegal, Project Officer, Office of Program Development, Air Pollution Control Office, Environmental Protection Agency, and S. J. Aresco, Coordinator, Energy Data Bank, Bureau of Mines, U. S. Department of the Interior.

We would also like to thank Jerrold G. Thompson, Branch

of Computer Sciences and Engineering, Bureau of Mines, U.S. Department of the Interior, Harry Buckley, Director, Maryland Geological Survey, and William E. Edmunds, Head Coal Geologist, Pennsylvania Bureau of Topographic and Geologic Survey for their assistance.

All unpublished data used in the project were supplied by the Air Pollution Control Office and the Bureau of Mines.

Cambridge, Massachusetts
April 1971

John C. Goodrich
Project Director and Editor

VOLUME TWO
GENERAL DOCUMENTATION
PART II

Table of Contents

PREFACE
TABLE OF CONTENTS
LIST OF ILLUSTRATIONS

1. INTRODUCTION
    1.1. Fundamentals of SYMAP for Contour Maps
    1.2. Proximal Map Considerations
    1.3. Conformant Map Considerations

2. PREPARATION OF PACKAGES
    2.1. Determining Locations on a Source Map
        2.1.1. Row and Column Measurement
        2.1.2. Equal Unit Measurement
        2.1.3. Use of "Legends", "Title", and "Text"
    2.2. A-Outline
    2.3. A-Conformolines
    2.4. B-Data Points
    2.5. C-Otolegends
        2.5.1. Characters and Symbols
        2.5.2. Types of Legends
            2.5.2.1. Point Legend
            2.5.2.2. Line Legend
            2.5.2.3. Area Legend
    2.6. D-Barriers
    2.7. E-Values
    2.8. E1-Values Index
    2.9. F-Map

3. MAP PACKAGE ELECTIVES
  - 3.1. Elective Number 1 - Size of Output Map (1 card)
  - 3.2. Elective Number 2 - Extreme Points of Input Map (1 card)
  - 3.3. Elective Number 3 - Number of Levels (1 card)
  - 3.4. Elective Number 4 - Value Range Minimum (1 card)
  - 3.5. Elective Number 5 - Value Range Maximum (1 card)
  - 3.6. Elective Number 6 - Value Range Intervals (1 or 2 cards)
  - 3.7. Elective Number 7 - Symbolism (5 cards)
  - 3.8. Elective Number 8 - Suppression of Contour Lines (1 card)
  - 3.9. Elective Number 9 - Suppression of Histogram Bars (1 card)
  - 3.10. Elective Number 10-Text (3 to 32 cards)
  - 3.11. Elective Number 11 - Printing Actual Value at Each Data Point (1 card)
  - 3.12. Elective Number 12 - Multiple Elective Repeat (1 card)
  - 3.13. Elective Number 13 - Scale (1 card)
  - 3.14. Elective Number 14 - Margin (1 card)
  - 3.15. Elective Number 15 - Number of Output Characters per Inch (1 card)
  - 3.16. Elective Number 16 - Large Size Maps (1 card)
  - 3.17. Elective Number 17 - Suppression of Tabular Output Data for Conformolines on Data Points (1 card)
  - 3.18. Elective Number 18 - Invalidation of Minus Zero and Blank Values. (1 card)
  - 3.19. Invalid Data Range Electives
    - 3.19.1. Elective Number 19 - Minimum Value for Invalid-Data Range (1 card)
    - 3.19.2. Elective Number 20 - Maximum Value for Invalid Data Range (1 card)
    - 3.19.3. Electives 19 and 20 in Combination (2 cards)

3.20. Elective Number 21 - Store Location Values for Map on Tape (1 card)
3.21. Elective Number 22 - Continuous Contours (1 card)
3.22. Elective Number 23 - Suppression of Invalid Data Point Symbol (1 card)
3.23. Elective Number 24 - Suppression of Numeric Interpretation (1 card)
3.24. Elective Number 25 - Suppression of Data Point Symbols (1 card)
3.25. Elective Number 26 - Overprint Alignment (1 card)
3.26. Elective Number 27 - Distinguishing Map Type (1 card)
3.27. Elective Number 28 - Dot Conformant Map (1 card) (available in SYMAP Version 6 only)
3.28. Extrapolation Range Electives
  3.28.1. Elective Number 31 - Relative Fractional Extrapolation (1 card)
  3.28.2. Elective Number 32 - Absolute Extrapolation Minimum (1 card)
  3.28.3. Elective Number 33 - Absolute Extrapolation Maximum (1 card)
3.29 Search Radius Electives
  3.29.1. Elective Number 34 - Initial Search Radius (1 card)
  3.29.2. Elective Number 35 - Maximum Search Radius (1 card)
3.30. Elective Number 36 - Number of Data Points for Interpolation (1 card)
3.31. Elective Number 37 - Independence (1 card)
3.32. Proximal Map Electives (3 cards)
3.33. Elective Number 38 - Data Graph (1 card) (Available in SYMAP Version 6 only)
3.34. Elective Number 39 - Linear Regression Plane (1 card) (Available in SYMAP Version 6 only)

4. SYMAP USER SUBROUTINE
  - 4.1. Multiple Map Submission
  - 4.2. Data Banks and Formats
  - 4.3. Manipulation of Coordinates

5. PROGRAMMING AND SYSTEMS GUIDE
  - 5.1. General Information
    - 5.1.1. Minimum Machine Configuration
    - 5.1.2. Program Storage Requirements
    - 5.1.3. Input and Output Units
    - 5.1.4. SYMAP Source Tape
  - 5.2. Programmer Information
    - 5.2.1. Program Times
    - 5.2.2. User Subroutine
    - 5.2.3. Blank COMMON
    - 5.2.4. Labeled COMMON
    - 5.2.5. Entry Statements
    - 5.2.6. Checking for missing Data
  - 5.3. System Information
    - 5.3.1. Overlay Structure
    - 5.3.2. Generation of Overlays
    - 5.3.3. Running Efficiency
    - 5.3.4. Program Execution
  - 5.4. Printer Information
    - 5.4.1. Spacing of Rows of Output
    - 5.4.2. Carriage Control Characters for Overprinting
    - 5.4.3. Printer Carriage Control Tape

6. THE "SYMVU" PROGRAM

- 6.1. Introduction
  - 6.1.1. Basic Principles
  - 6.1.2. Required Input
- 6.2. Elective Card 2
  - 6.2.1. Elective 2-1: Rows
  - 6.2.2. Elective 2-2: Columns
  - 6.2.3. Elective 2-3: View
  - 6.2.4. Elective 2-4: Line Type
  - 6.2.5. Elective 2-5: Smooth
  - 6.2.6. Elective 2-6: Repeat
  - 6.2.7. Elective 2-7: Lnint
  - 6.2.8. Elective 2-8: Nozero
  - 6.2.9. Elective 2-9: Sqrtt
  - 6.2.10. Elective 2-10: Sameh
  - 6.2.11. Elective 2-11: Endlin
  - 6.2.12. Elective 2-12: Symap
  - 6.2.13. Elective 2-13: Leg
  - 6.2.14. Elective 2-14: Nscal
  - 6.2.15. Elective 2-15: Flp
  - 6.2.16. Elective 2-16: Base
  - 6.2.17. Elective 2-17: Pltdot
  - 6.2.18. Elective 2-18: Fdata
  - 6.2.19. Elective 2-19: Namin
  - 6.2.20. Elective 2-20: Namax
  - 6.2.21. Elective 2-21: Iform
  - 6.2.22. Elective 2-22: Nleg
- 6.3. Elective Card 3
  - 6.3.1. Elective 3-1: Altitude
  - 6.3.2. Elective 3-2: Azimuth
  - 6.3.3. Elective 3-3: Width
  - 6.3.4. Elective 3-4: Height
  - 6.3.5. Elective 3-5: Amin
  - 6.3.6. Elective 3-6: Amax
  - 6.3.7. Elective 3-7: Vdist
  - 6.3.8. Elective 3-8: Step
  - 6.3.9. Elective 3-9: Sepsm
  - 6.3.10. Elective 3-10: Size
- 6.4. Optional Data Cards
  - 6.4.1. Data Matrix on cards
  - 6.4.2. Symbol Points
  - 6.4.3. Legends
- 6.5. Subroutine Data
- 6.6. Sample Submissions
- 6.7. Technical Information
  - 6.7.1. SYMVU Source Tape Description
  - 6.7.2. Minimum Machine Configuration
  - 6.7.3. Units Description
  - 6.7.4. Input
  - 6.7.5. Calcomp Plotter Routines Used in SYMVU

7. THE "GRID" PROGRAM

- 7.1. Introduction
  - 7.1.1. Basic Principles
  - 7.1.2. Required Input
- 7.2. Data Package
  - 7.2.1. Data Option A
  - 7.2.2. Data Option B
  - 7.2.3. Assigning Data to Map Levels
  - 7.2.4. Multiple Data Sets
- 7.3. Map Package
  - 7.3.1. Standard Format
  - 7.3.2. Elective 1: Grid (1 card)
  - 7.3.3. Elective 2: Data (1 card)
  - 7.3.4. Elective 3: Number of Levels (1 card)
  - 7.3.5. Elective 4: Value Range Minimum (1 card)
  - 7.3.6. Elective 5: Value Range Maximum (1 card)
  - 7.3.7. Elective 6: Value Range Intervals (1 to 2 cards)
  - 7.3.8. Elective 7: Symbolism (5 cards)
  - 7.3.9. Elective 8: Flag Point (1 card)
  - 7.3.10. Elective 9: Histogram (1 card)
  - 7.3.11. Elective 10: Text (3-32 cards)
  - 7.3.12. Elective 11: Data Record (1 card)
  - 7.3.13. Elective 12: Dot Map (1 card)
  - 7.3.14. Elective 13: Grid Numbering (1 to 2 cards)
  - 7.3.15. Elective 14: Prescaled Data (1 card)
  - 7.3.16. Elective 15: Timer (1 card)
  - 7.3.17. Elective 16: (only available in a special version of the program) Legends (1 card)
- 7.4. Irregular Outlines
  - 7.4.1. Filling in the Rectangle
  - 7.4.2. Irregular Outline Package
- 7.5. Subroutine Flexin
- 7.6. Computer Submissions

8. PROGRAM MODIFICATIONS

- 8.1. Print Out Statements in SYMAP
- 8.2. Legends With GRID
- 8.3. C-LEGENDS Package

List of Illustrations

PART II: COMPUTER PROGRAM MANUALS

Example

1-1	Contour Source Map
1-2	Source Map With Legends
1-3	Input Listing: Contour By Row and Columns
1-4	Output Listing: Contour By Row and Columns
1-5	Contour Map: Rows and Columns
1-6	Input Listing: Contour Map
1-7	Output Listing: Contour Map
1-8	Contour Map 1: Standard Electives
1-9	Input Listing: Contour Map Packages
1-10	Contour Map 2: Maximum and Minimum Values
1-11	Contour Map 3: Value Range Intervals
1-12	Contour Map 4: Symbolism
1-13	Contour Map 5: Inset Area
1-14	Input Listing: Proximal Map
1-15	Output Listing: Proximal Map
1-16	Proximal Map 1: Standard Electives
1-17	Proximal Map 2: Special Symbolism
1-18	Zone Source Map
1-19	Zone Source Map With Conformant Boundaries
1-20	Input Listing: Conformant Map
1-21	Output Listing: Conformant Map
1-22	Conformant Map

Table 2-1 Specifications for Otolegends

Example

2-1	Point Legend
2-2	Line Legend
2-3	Area Legend

Example
3-1 Source Map
3-2 Elective 2
3-3 Electives 3 through 6
3-4 Elective 6 - 5 levels
3-5 Elective 6 - 4 levels
3-6 Elective 6 - 7 levels
Table 3-1 Standard Symbolism for Various Levels
Table 3-2 Symbolism for Levels and Special Purposes
Example
3-7 Elective 7 - 6 levels
3-8 Elective 7 - contour lines black
3-9 Elective 7 - proximal
3-10 Elective 12

4-1 Subroutine FLEXIN--Dummy
4-2 Subroutine FLEXIN--Non-Standard Format
4-3 Subroutine FLEXIN--Manipulation and Data Storage
4-4 Subroutine FLEXIN--Dimensioning and Data Points
4-5 Subroutine FLEXIN--Axes Transformation

5-1 Utility Program to Punch Source Card from SYMAP Tape
5-2 Overlay Tree
5-3 Control Cards to Link Edit Object Modules into Load Modules as the Overlay Segments
5-4 Compile and Link Edit SYMAP into Overlay Segment Load Modules
5-5 SYMAP submission at the LCG
5-6 Job Control Language to Load and Execute SYMAP
5-7 Job Control Language to Override SYMAP subroutines with edited subroutines

Example:
6-1 Azimuth Orientation System

Table 6-1 Symbol Points
Table 6-2 Electives Used in Examples

Examples:
6-2 Figure A: Mantegna Bay - Conformant
6-3 Figure B: Mantegna Bay - Contour
6-4 Figure C: Mantegna Bay - Proximal
6-5 Figure 2: Mantegna Bay - Isometric Projection
6-6 Figure 3: Mantegna Bay - Two Point Perspective
6-7 Figure 4: Mantegna Bay - Planometric Projection
6-8 Listing of Electives and Their Uses
6-9 Utility Program to Punch Source Cards from SYMVU Tape
6-10 Listing of Job Control Language to Load Binary Deck of SYMVU onto Disk
6-11 Listing of Job Control Language to run SYMVU sample input
6-12 Harvard's Catalogued Procedure for SYMVU

7-1 Equal Value Range Levels
7-2 Unequal Value Range Levels/Four Levels
7-3 Unequal Value Range Levels/ Seven Levels
7-4 Level Symbol Specification
7-5 Grid Numbering Subdivided Grid
7-6 Irregular Outline Package
7-7 Simplest Use of Subroutine Flexin
7-8 Multiple Uses of Subroutine Flexin
7-9 Sample Submission for Four Maps

8-1 SYMAP Print Out Changes
8-2 SYMAP Changes for Legends
8-3 GRID Changes for Legends
8-4 C-LEGENDS Specifications

II. COMPUTER PROGRAM MANUALS

1. Introduction to SYMAP
2. Preparation of Packages
3. Map Package Electives
4. SYMAP User Subroutine
5. Programming and Systems Guide
6. The SYMVU Program
7. The GRID Program
8. Program Modifications

1. INTRODUCTION TO SYMAP

SYMAP is a computer program for producing maps which graphically depict spatially arrayed quantitative and qualitative information. It is suited to a broad range of applications, and has numerous options for meeting widely varying requirements. Combinations of standard computer characters are used to display a scale of tone from black to white which corresponds to the data value range. Mathematical computations can be performed on the data, if necessary, before mapping. Options are provided for the rescaling of raw data and for variations in symbolism.

By assigning values to the coordinate locations of data points or data zones, one or more of three basic types of map may be produced, as specified by the user:

---CONTOUR: (based on the use of contour lines, each of which represents a uniform value throughout its length). The contour (or isoline) map consists of closed curves known as contour lines which connect all points having the same numeric value or height. Contour lines emerge from a datum plane at selected levels which are determined from the scale of the map and the range of the data. Between any two contour lines, a continuous variation is assumed. The surface is generated by interpolation between values at given data points. The contour lines are then drawn on this interpolated surface.

---CONFORMANT: (based on conformance to the boundaries of a data zone)
The conformant map is best suited for data, either qualitative or quantitative, whose areal limits are of significance,

and whose representation as a continuous surface is inappropriate. Each data zone is enclosed by a boundary "conformant" to some predefined spatial unit. The entire spatial unit is given the same value, and symbolism is assigned according to its numeric class.

---PROXIMAL: (based on proximity to a data point). The proximal map is very similar in appearance to the conformant map. However, the spatial units are defined by nearest neighbor methods from point information. Each character location on the output map is assigned to the value of the data point nearest to it. Boundaries are assumed along the line where the values change and conformant mapping is applied.

To produce any of the three types of computer maps, a deck of punched cards must be prepared as input to the computer. This deck consists of a number of "packages" each covering a specific category of information. These packages, are listed below; they are explained in detail in the section Preparation of Packages.

The MAP package specifies, below the map, an appropriate title for the identification of each separate map that is to be run. It then instructs the computer to make each map by using certain electives, as required. The availability of these electives is a major feature of the SYMAP program and adds greatly to its flexibility. Some of the electives are:

(i) Size: used to specify, by inches and decimal fractions thereof, the overall dimensions of the rectangular map border surrounding the study area.

(ii) Content: enables the computer to change the scale of the map to be produced or to map a portion of the study area at the scale originally specified or any other scale desired.

(iii) Number of Levels: used to specify the number of levels and/or class intervals (not more than 12) into which the total range of the data is to be subdivided for mapping purposes.

(iv) Histogram Bars: used to suppress the normally-provided bars whose length show the frequency distribution of the data.

(v) Text: used below the map to provide supplementary information.

Other electives deal with such subjects as symbols, maximum and minimum values, borders, and scale. The MAP package is always required; use of the other packages is determined by the type of map desired.

The OUTLINE package describes the outline of the study area boundary by specifying the coordinate locations of the vertices.

The DATA POINTS package is used to give the positions of the points for which data are available, by specifying their coordinate locations. The locations may be either points or centers of areas ("data zones") for which data are available.

The OTOLEGENDS package causes supplementary information to appear on the face of the map by specifying the coordinate locations and content. Legends include borders, location names and numbers, and a title block.

The VALUES package assigns numerical data to the data points, by specifying the values involved. Raw data given to the computer may be related, manipulated, weighted, and aggregated in any manner desired by means of a user subroutine.

1.1. FUNDAMENTALS OF SYMAP FOR CONTOUR MAPS

The following sections are designed as an introduction to the specifications for preparing the various packages and map package electives described in the next two portions of the manuals. The illustrations show nearly every aspect of the information needed to produce a contour, conformant, or proximal map.

The first step is to define an area on a source map using some x-y reference system (see Example 1-1). The origin should be in the upper left hand corner since the computer is accustomed to reading down rather than up. Study area boundaries, legends, and data points or zone centers should all be coded using this x-y reference system. Examples of point, text, line, and area legends are shown in Example 1-2. Two separate outlines (or islands) are evident in Example 1-1 (a to f, and g to j) and there are six data point locations.

The A-OUTLINE, B-DATA points, and C-OTOLEGENDS packages that would be coded from this information, together with an F-MAP package which is always required, can be used to produce a base map for testing purposes. The E-VALUES package--which must have one to one correspondence with the B-DATA POINTS package so that the correct value is associated with the right location--is then added to produce a contour map. A surface is interpolated at all points, based on the values at the given data points.

Example 1-3 shows the input--measured by rows and columns*--which produced the output in Example 1-4 and contour map in Example 1-5. Example 1-6 shows the input--measured in inches--which produced the output in Example 1-7 and contour map in Example 1-8. The maps in Example 1-5 and Example 1-8 are the same except for (1) slight variations in the location of outline and data points, and (2) additional legends in Example 1-8. The material below the map is exactly the same and, therefore, not included with Example 1-8. It is recommended that equal unit measurement (such as inches) be used for most applications.

Example 1-9 shows the various F-MAP packages used to produce the five contour maps in Examples 1-8, 1-10, 1-11, 1-12, and 1-13. Packages which are not changed from map to map remain in effect; therefore, as specified in Example 1-9, the same A-OUTLINE, B-DATA POINTS, C-OTOLEGENDS, and E-VALUES packages would be used for all five maps. A new F-MAP package, however, must be specified for each map.

Electives 1 thru 10 and the repeat elective, 12, are all demonstrated by these examples. These are the most common electives used and should be examined carefully. The extent of the source map to be displayed (Elective 2) should be determined before the size of the output map (Elective 1) is specified. Note that some of the legends change size with the scale in Example 1-13.

---

* These are the rows and columns of computer printout as found on the output map; there are 8 rows per inch vertically and 10 columns per inch horizontally. This is discussed further under Preparation of Packages.

1.2 PROXIMAL MAP CONSIDERATIONS

The only difference to the user in specifying the input for a proximal map as opposed to a contour map is the use of electives 31, 36, and 37 in the F-MAP package. Example 1-14 shows the input for producing the two proximal maps in Examples 1-16 and 1-17. Example 1-15 shows some of the output preceding the map in Example 1-16. Example 1-14 illustrates several new features:

(i) the use of elective 11 to print the actual values on the map in Example 1-17;

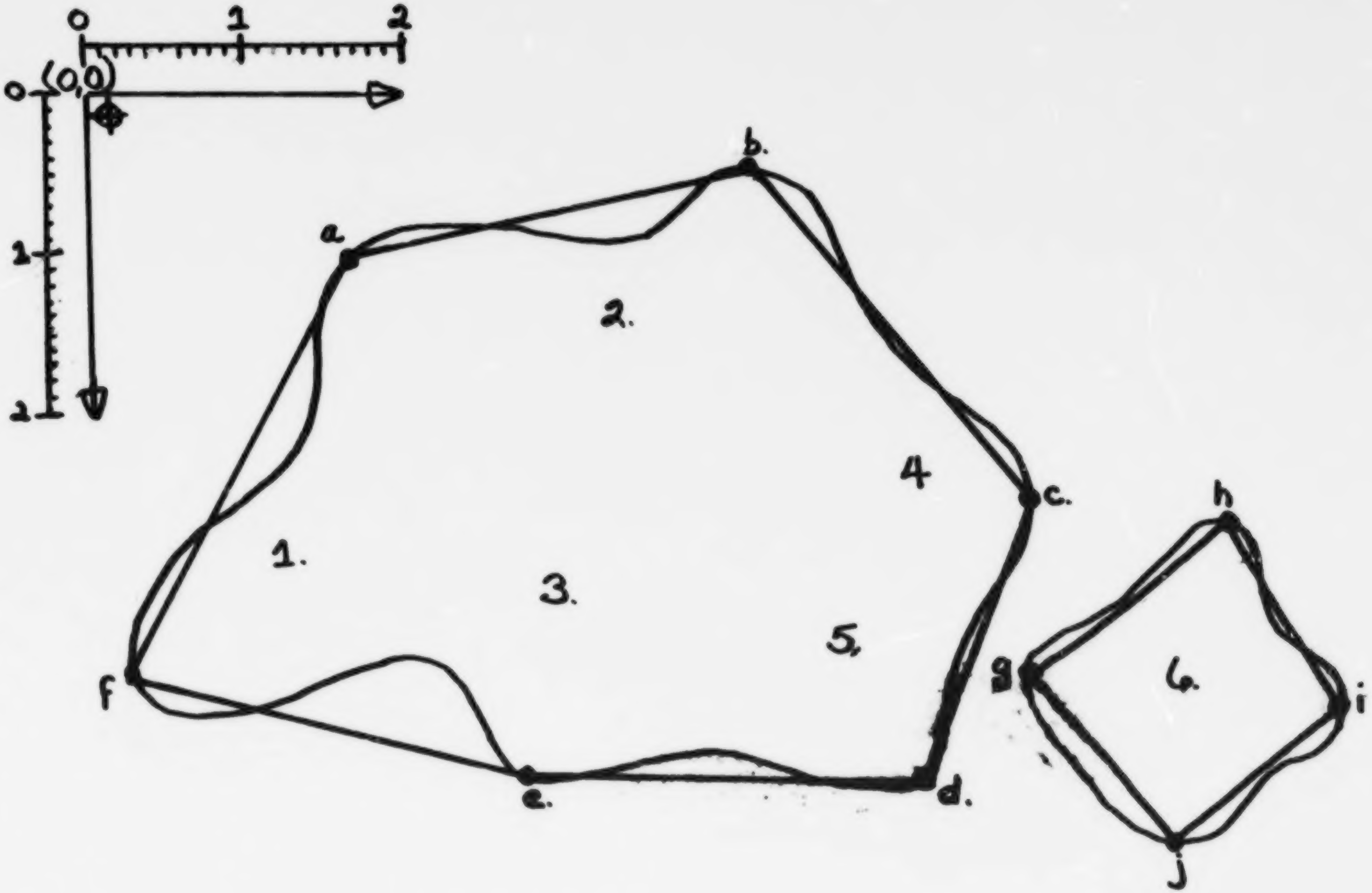
(ii) running more than one map in a single submission with the deletion of a package (C-OTOLEGENDS) from the first to the second map; and,

(iii) the use of a 'six nines' card to signal that all packages have been input.

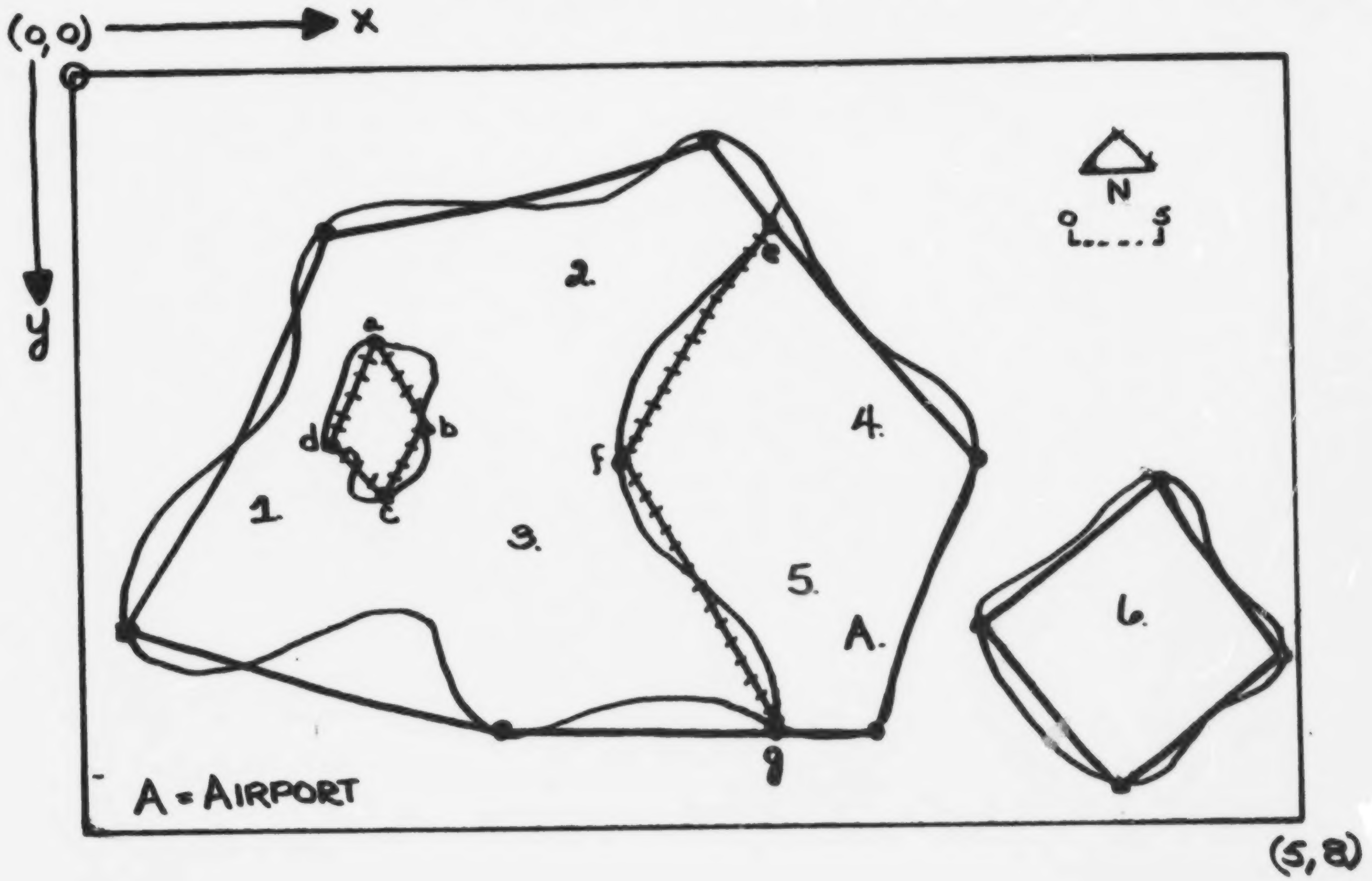
Notice that the outline, data points, and legends packages are the same as those shown in Example 1-6.

1.3 CONFORMANT MAP CONSIDERATIONS

Example 1-18 shows the source map used for the conformant map, and Example 1-19 adds the zone boundaries as they would be coded in the A-CONFORMOLINES package. The input listing in Example 1-20 produced the output in Example 1-21 and conformant map in Example 1-22. Note how the symbolism completely fills each zone as defined in the conformolines package. The A-CONFORMOLINES package takes the place of both the outline and data points packages used with contour and proximal maps. There should be a one to one correspondence between the conformant zones and the values associated with them.


0
1
2
(0,0)
0
1
2
a
b
c.
d.
e.
f
g
h
i
j
1.
2.
3.
4
5.
6.

Example 1-1
Contour Source Map


(0,0)
x
y
a
b
c
d
e
f
g
1.
2.
3.
4.
5.
6.
A.
N
0
5
A = AIRPORT
(5,8)

Example 1-2
Source Map with Legends

```
A-OUTLINE                            8.            10.
                   9.0          16.0
                   4.0          41.0
                  21.0          59.0
                  36.0          52.0
                  35.0          27.0
                  30.0           2.0
                   9.0          16.0
                  30.0          58.0
                  23.0          71.0
                  32.0          78.0
                  39.0          68.0
                  30.0          58.0
99999
B-DATA POINTS                        8.            10.
                  24.0          13.0
                  13.0          34.0
                  26.0          30.0
                  20.0          53.0
                  28.0          48.0
                  30.0          70.0
99999
C-OTOLEGENDS                         8.0           10.
   11     P       37.0           2.0
A = AIRPORT
     A    P       31.0          52.0
99999
E-VALUES
                  12.0
                 -14.0
                  43.0
                  38.0
                  -4.0
                   0.0
99999
F-MAP
TEST MAP #1
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 197C
ROW AND COLUMN MEASUREMENT
    1              5.0           8.0
    2              0.0           0.0           5.0          8.0
99999
```

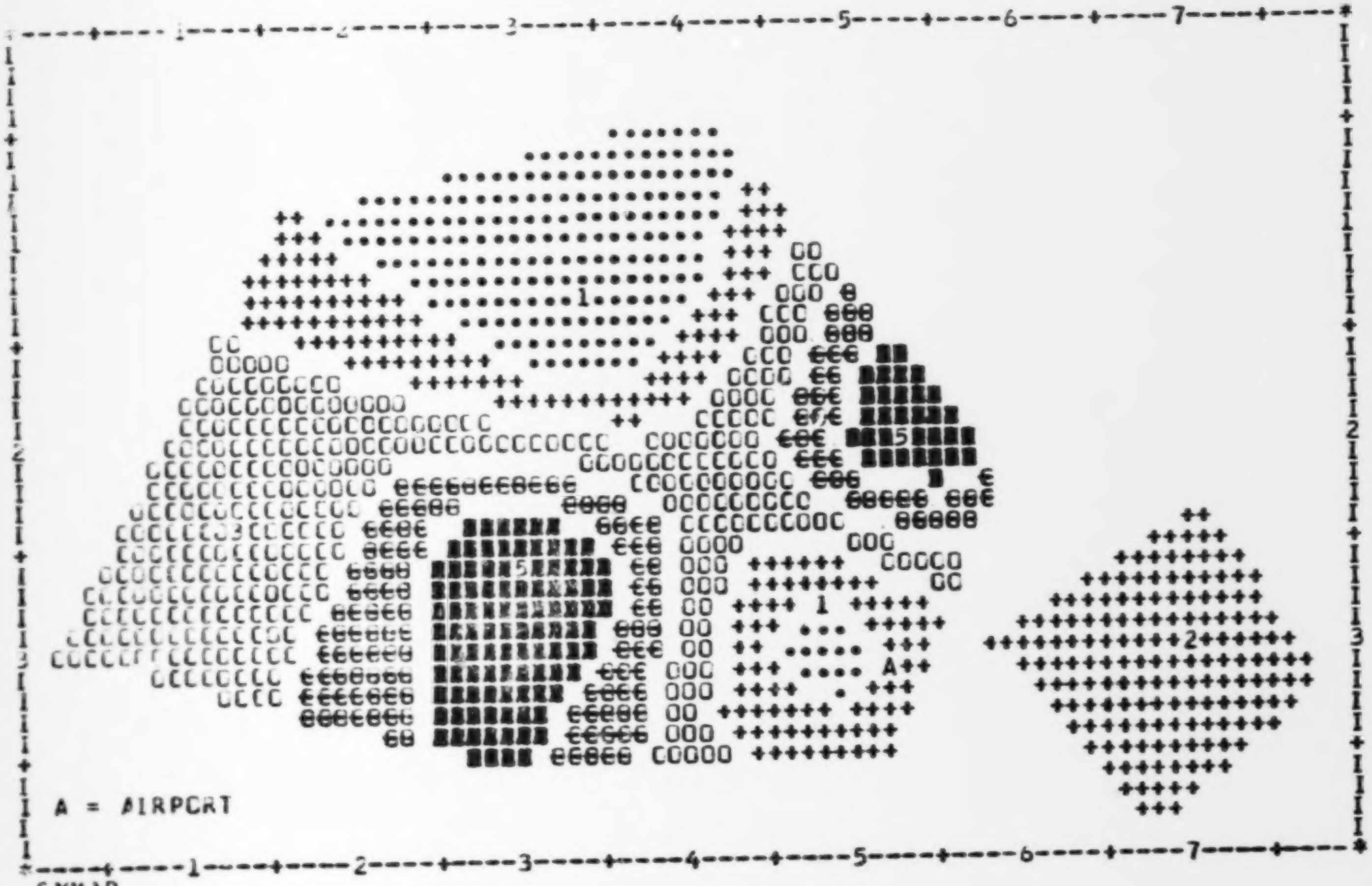
Example 1-3
Input Listing: Contour By Row And Columns

```
A-OUTLINE
---------

DOWN COORDINATE DIVIDED INTERNALLY BY       8.00
ACROSS COORDINATE DIVIDED INTERNALLY BY     10.00

VERTEX              DOWN     ACROSS
                   ISLAND    1

(    1)             9.00     16.00
(    2)             4.00     41.00
(    3)            21.00     59.00
(    4)            36.00     52.00
(    5)            35.00     27.00
(    6)            30.00      2.00
AREA=     1239.00            CENTER=(       21.63,       32.26)
                   ISLAND    2

(    1)            30.00     58.00
(    2)            23.00     71.00
(    3)            32.00     78.00
(    4)            39.00     68.00
AREA=      163.00            CENTER=(       30.95,       68.51)


B-DATA POINTS
-------------

DOWN COORDINATE DIVIDED INTERNALLY BY       8.00
ACROSS COORDINATE DIVIDED INTERNALLY BY     10.00

POINT               DOWN     ACROSS

(    1)            24.00      13.00
(    2)            13.00      34.00
(    3)            26.00      30.00
(    4)            20.00      53.00
(    5)            28.00      48.00
(    6)            30.00      70.00


C-OTOLEGENDS
------------

DOWN COORDINATE DIVIDED INTERNALLY BY       8.00
ACROSS COORDINATE DIVIDED INTERNALLY BY     10.00
VERTEX              DOWN     ACROSS     +ROWS      +COLS
(    1) 'A = AIRPORT' ACROSS FROM
                   37.00      2.00        0.         0.
(    2) 'A' AT POINT

                   31.00     52.00        0.         0.
```

Example 1-4
Output Listing: Contour by Row And Columns

```
E-VALUES
--------


DATUM              VALUE

(    1)            12.00
(    2)           -14.00
(    3)            43.00
(    4)            38.00
(    5)            -4.00
(    6)             0.0


F-MAP
-----

TEST MAP #1

CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970

ROW AND COLUMN MEASUREMENT

ELECTIVE

     1 MAP SIZE IS              5.00 INCHES LONG BY              8.00 INCHES WIDE
     2 EXTREME POINTS ARE (        0.0 ,        0.0 ) AND (        5.00,        8.00)
TIME =     0.0



MAP    1
-------

TEST MAP #1
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
ROW AND COLUMN MEASUREMENT




MAP SCALE =       1.0000 INCHES ON OUTPUT MAP/UNITS ON SOURCE MAP

MAP SHOULD BE PRINTED AT    8.0 ROWS PER INCH AND  10.0 COLUMNS PER INCH

ROW    = DOWN   COORDINATE *      8.0000
COLUMN = ACROSS COORDINATE *     10.0000


DATA POINTS FOR MAP

POINT          ROW     COLUMN        DATUM        VALUE        LEVEL

   1)           24         13            1        12.00            3
   2)           13         34            2       -14.00            1
   3)           26         30            3        43.00            5
   4)           20         53            4        38.00            5
   5)           28         48            5        -4.00            1
   6)           30         70            6         0.0             2
STANDARD SEARCH RADIUS IS            3.9271

TIME =     0.0
```

Example 1-4
(Continued)


A = AIRPORT
1
2
3
4
5
6
7

```
SYMAP
TIME =    0.0

TEST MAP #1
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
ROW AND COLUMN MEASUREMENT
```

Example 1-5
Contour Map: Rows And Columns

```
DATA VALUE EXTREMES ARE       -14.00       43.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
      ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)

  MINIMUM     -14.00      -2.60       8.80      20.20      31.60
  MAXIMUM      -2.60       8.80      20.20      31.60      43.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

              20.00      20.00      20.00      20.00      20.00


FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL          1          2          3          4          5
            ====================================================
             .........  +++++++++  000000000  ΘΘΘΘΘΘΘΘΘ  █████████
             .........  +++++++++  000000000  ΘΘΘΘΘΘΘΘΘ  █████████
  SYMBOLS    ....1....  ++++2++++  000030000  ΘΘΘΘ4ΘΘΘΘ  ████5████
             .........  +++++++++  000000000  ΘΘΘΘΘΘΘΘΘ  █████████
             .........  +++++++++  000000000  ΘΘΘΘΘΘΘΘΘ  █████████
            ====================================================
  FREQ.          2          1          1          0          2
         1    I..1..I    I++2++I    I00300I               I██5██I
         2    I..1..I                                     I██5██I

TIME =     0.0
```

Example 1-5
(Continued)

```
A-OUTLINE
                         1.0          1.6
                          .5          4.1
                         2.5          5.8
                         4.3          5.2
                         4.3          2.6
                         3.6           .2
                         1.0          1.6
                         3.5          5.8
                         2.6          7.1

                         3.8          7.8
                         4.6          6.8
                         3.5          5.8
99999
B-DATA POINTS
                         2.9          1.3
                         1.5          3.4
                         3.1          3.0
                         2.4          5.3
                         3.4          4.8
                         3.7          7.0
99999
E-VALUES
                        12.0
                       -14.0
                        43.0
                        38.0
                        -4.0
                         0.0
99999
```

Example 1-6
Input Listing: Contour Map

```
C-OTOLEGENDS
    3    P         .6       6.8                -1.0
***
    5    P         .6       6.8       1.0      -2.0
*****
    1    P         .6       6.8      -1.0
*
    1    P         .6       6.8       2.0
N
    6    P         .6       6.8       5.0      -2.0
SCALE
     -   L        1.3       6.3
                  1.3       7.3
    1    P        1.3       6.3
|
    1    P        1.3       7.3
|
    1    P        1.3       6.8
+
    1    P        1.3       6.3      -1.0
0
    1    P        1.3       6.8      -1.0
5
    2    P        1.3       7.3      -1.0
10
   11    P        4.6        .2
A = AIRPORT
         A        1.8       2.0
                  2.3       2.2
                  2.8       2.0
                  2.5       1.5
                  1.8       2.0
     A   P        3.8       5.2
     +=  L        4.4        .2
                  4.4        .4
    7    P        4.4        .6
= CANAL
     +=  L        1.0       4.6
                  2.5       3.5
                  4.4       4.5
99999
E-MAP
TEST MAP =2
CONTOUR MAP - ( CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
    1             5.0       8.0
    2             0.0       0.0       5.0       8.0
99999
```

Example 1-6
(Continued)

```
A-OUTLINE
---------


VERTEX              DOWN    ACROSS
                   ISLAND    1

(    1)             1.00     1.60
(    2)             0.50     4.10
(    3)             2.50     5.80
(    4)             4.30     5.20
(    5)             4.30     2.60
(    6)             3.60     0.20
AREA=      15.05              CENTER=(       2.61,        3.17)
                   ISLAND    2

(    1)             3.50     5.80
(    2)             2.60     7.10
(    3)             3.80     7.80
(    4)             4.60     6.80
AREA=       2.05              CENTER=(       3.61,        6.85)
```

```
B-DATA POINTS
-------------


POINT               DOWN    ACROSS

(    1)             2.90     1.30
(    2)             1.50     3.40
(    3)             3.10     3.00
(    4)             2.40     5.30
(    5)             3.40     4.80
(    6)             3.70     7.00
```

```
E-VALUES
--------


DATUM               VALUE

(    1)             12.00
(    2)            -14.00
(    3)             43.00
(    4)             38.00
(    5)             -4.00
(    6)              0.0
```

Example 1-7
Output Listing: Contour Map

```
C-OTOLEGENDS
------------

VERTEX              DOWN     ACROSS      +ROWS       +COLS
(    1) '***' ACROSS FROM
                    0.60        6.80        0.         -1.
(    2) '******' ACROSS FROM
                    0.60        6.80        1.         -2.
(    3) '*' AT POINT

                    0.60        6.80       -1.          0.
(    4) 'N' AT POINT

                    0.60        6.80        2.          0.
(    5) 'SCALE ' ACROSS FROM
                    0.60        6.80        5.         -2.
(    6) '-' ON LINE

(    1)             1.30        6.30
(    2)             1.30        7.30
LENGTH=         1.00
(    7) '|' AT POINT

                    1.30        6.30        0.          0.
(    8) '|' AT POINT

                    1.30        7.30        0.          0.
(    9) '+' AT POINT

                    1.30        6.80        0.          0.
(   10) '0' AT POINT

                    1.30        6.30       -1.          0.
(   11) '5' AT POINT

                    1.30        6.80       -1.          0.
(   12) '10' ACROSS FROM
                    1.30        7.30       -1.          0.
(   13) 'A = AIRPORT' ACROSS FROM
                    4.60        0.20        0.          0.
```

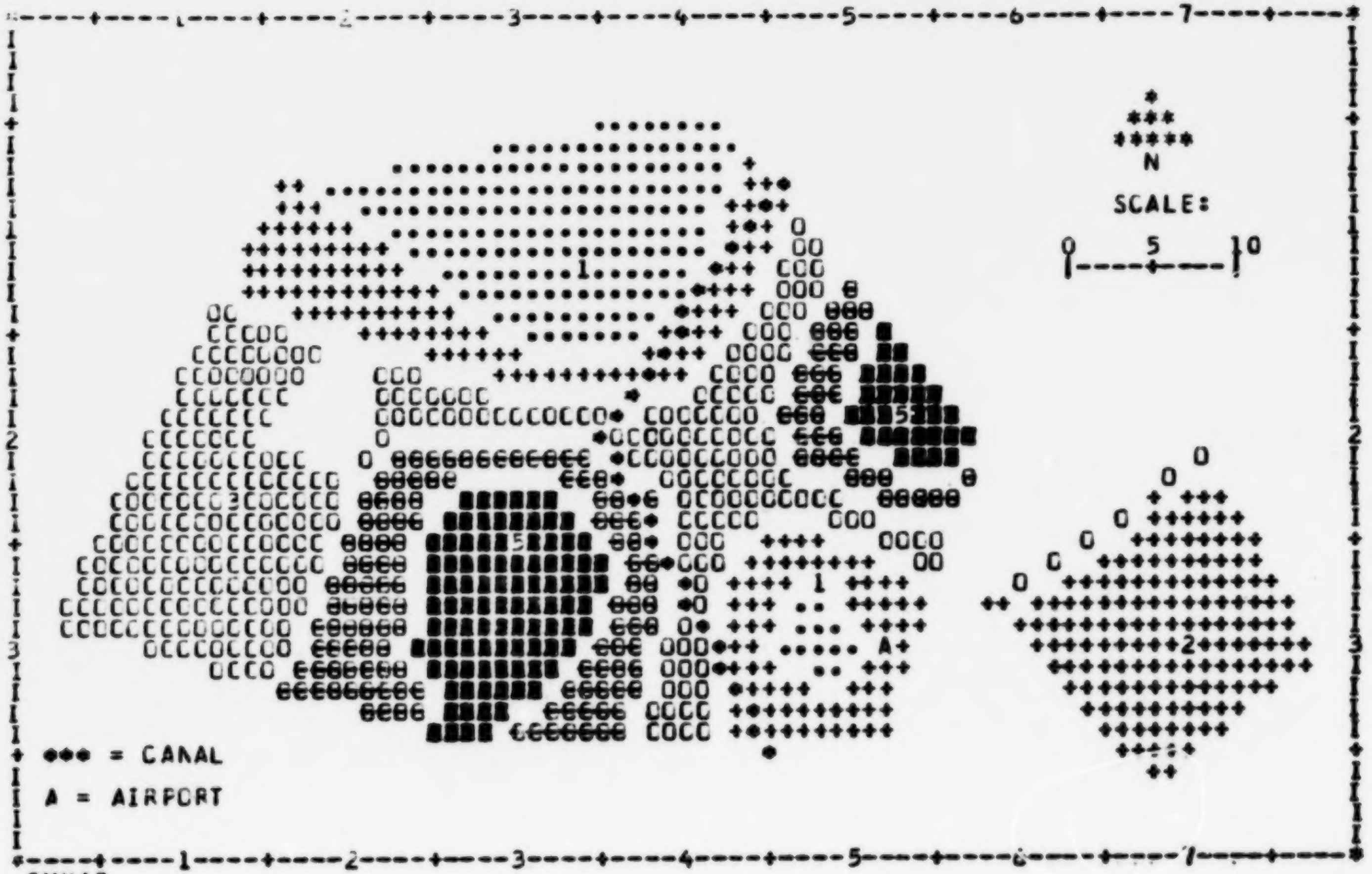
Example 1-7
(Continued)

```
(   14) ' ' IN AREA

(    1)            1.80      2.00
(    2)            2.30      2.20
(    3)            2.80      2.00
(    4)            2.50      1.50
(    5)            1.80      2.00
LENGTH=        2.52
AREA=       0.35
CENTER=(        2.35,     1.90)

(   15) 'A' AT POINT


                   3.80      5.20          0.        0.

(   16) '◆' CN LINE

(    1)            4.40      0.20
(    2)            4.40      0.40
LENGTH=       0.20

(   17) '= CANAL' ACROSS FROM

                   4.40      C.6C          0.        0.

(   18) '◆' ON LINE

(    1)            1.00      4.60
(    2)            2.50      3.50
(    3)            4.40      4.5C
LENGTH=       4.01
```

Example 1-7
(Continued)

```
F-MAP
-----

TEST MAP #2

CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970

MEASUREMENT IN INCHES

ELECTIVE

    1 MAP SIZE IS             5.00 INCHES LONG BY            8.00 INCHES WIDE
    2 EXTREME POINTS ARE (       0.0 ,       0.0 ) AND (       5.00,        8.00)

TIME =     0.0



MAP   2
-------

TEST MAP #2
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES




MAP SCALE =      1.0000 INCHES ON OUTPUT MAP/UNITS ON SOURCE MAP

MAP SHOULD BE PRINTED AT    8.0 ROWS PER INCH AND  10.0 COLUMNS PER INCH

ROW    = DOWN   COORDINATE *      8.0000
COLUMN = ACROSS COORDINATE *     10.0000


DATA POINTS FOR MAP

POINT          ROW     COLUMN       DATUM      VALUE       LEVEL

   1)           23        13           1      12.00           3
   2)           12        34           2     -14.00           1
   3)           25        30           3      43.00           5
   4)           19        53           4      38.00           5
   5)           27        48           5      -4.00           1
   6)           30        70           6       0.0            2

STANDARD SEARCH RADIUS IS              3.9958

TIME =     0.0
```

Example 1-7
(Continued)


N
SCALE:
0
5
10
●●● = CANAL
A = AIRPORT
1
2
3
4
5
6
7

SYMAP

TIME =   0.0

TEST MAP #2

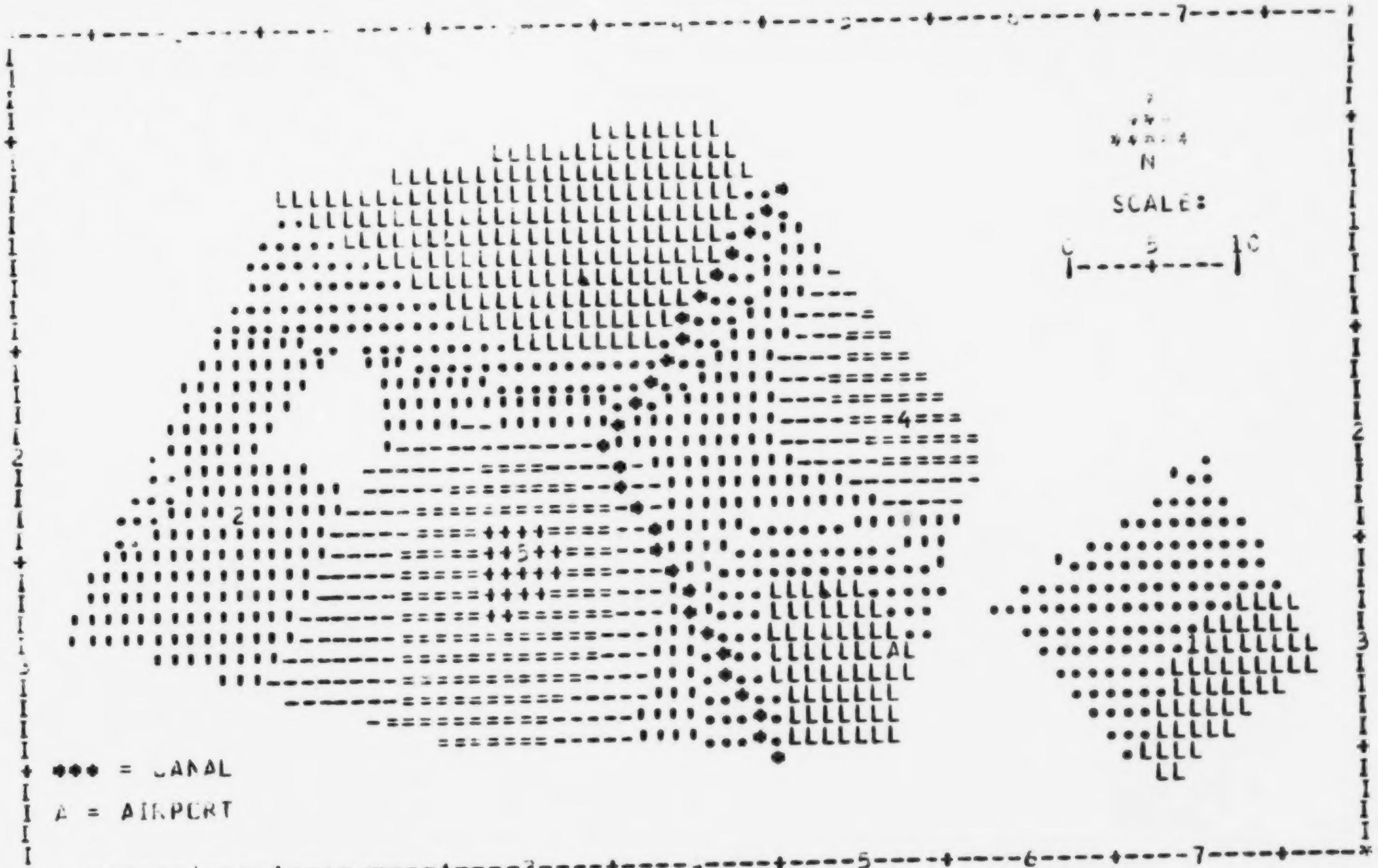
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970

MEASUREMENT IN INCHES

Example 1-8
Contour Map 1: Standard Electives

```
F-MAP
TEST MAP #2
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
    1            5.0       8.0
    2            0.0       0.0       5.0       8.0
99999
F-MAP
TEST MAP #4
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
    1            5.0       8.0
    2            0.0       0.0       5.0       8.0
    3           10.0
    4            0.0
    5          100.0
    8
   10
ALL AREAS WITH DECREASED POPULATION IN 1960-1970 ARE SHOWN BY THE
LOWEST LEVEL
9999
99999
F-MAP
TEST MAP #5
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
    1SAME
    2SAME
    4SAME
    5           40.0
    6            5.0       5.0      10.0      10.0      10.0
    8SAME
    9
99999
F-MAP
TEST MAP #6
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
   12
    3            6.0
    5          100.0
    6           10.0      10.0      10.0      10.0      10.0      50.0
    7
.+XOOO    123456    LLHH
   -XX               . H
     A                 H
     V                 /
99999
F-MAP
TEST MAP #7
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970

   12
    2            0.0       4.0       2.5       8.0
99999
```

Example 1-9
Input Listing: Contour Map Packages



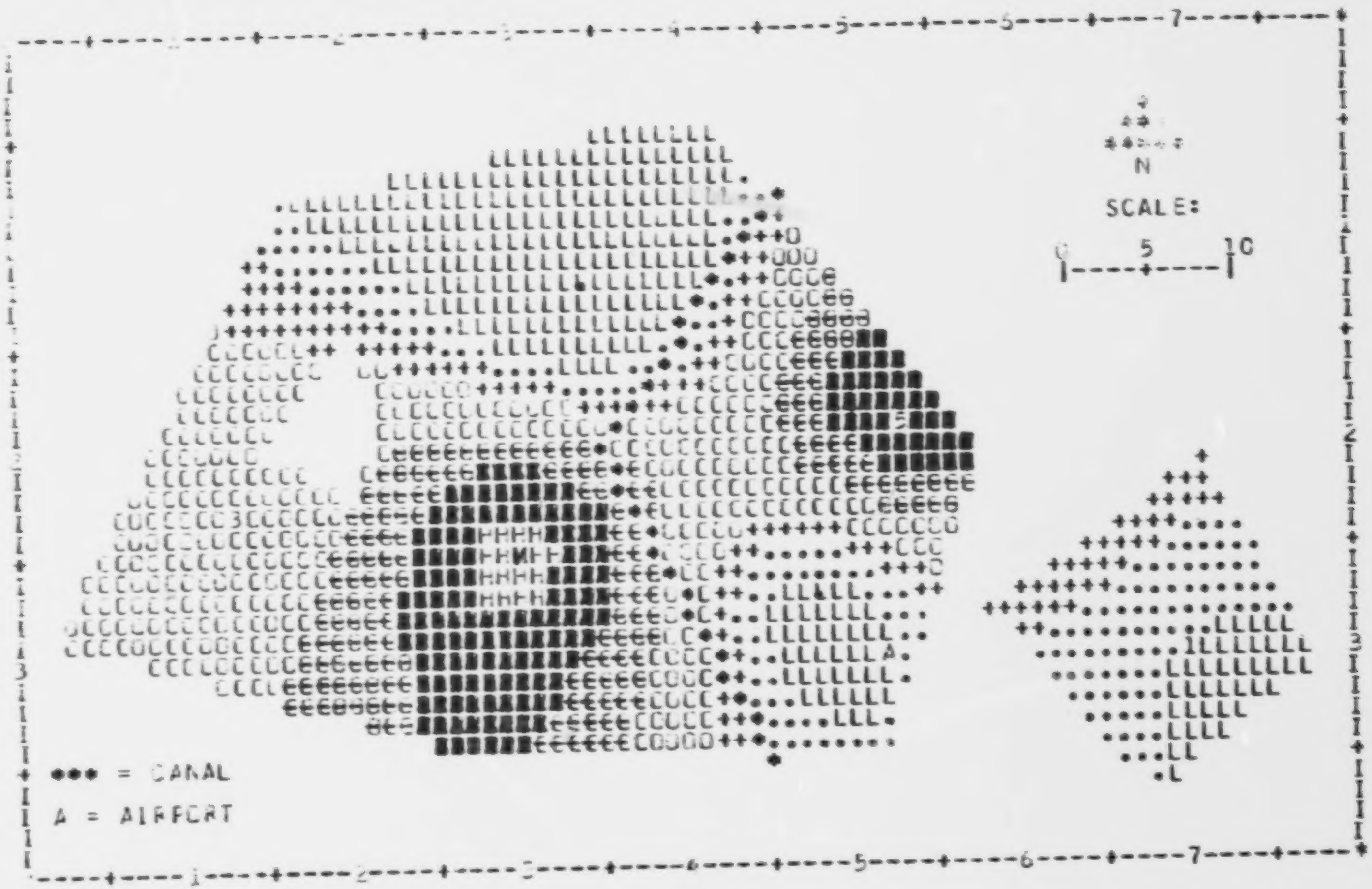
N
SCALE:
0 5 10
●●● = CANAL
A = AIRPORT
1
2
3
4
5
6
7

```
SYMAP
TIME =     0.0

TEST MAP #[illegible]
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES

                                        ALL AREAS WITH DECREASED POPULATION
                                        LOWEST LEVEL
```

Example 1-10
Contour Map 2: Maximum And Minimum Values

```
DATA VALUE EXTREMES ARE        -14.00       43.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
      ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)

  MINIMUM      BELOW       0.0       10.00       20.00       30.00       40.00       50.00
  MAXIMUM       0.0       10.00      20.00       30.00       40.00       50.00       60.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

                          10.00      10.00       10.00       10.00       10.00       10.00

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL          L          1          2          3          4          5          6         90.00
           ==========================================================================       100.00
            LLLLLLLLL  .........  '''''''''  ---------  =========  +++++++++  XXXXXXXXX
            LLLLLLLLL  .........  '''''''''  ---------  =========  +++++++++  XXXXXXXXX
  SYMBOLS   LLLLLLLLL  ....1....  ''''2''''  ----3----  ====4====  ++++5++++  XXXX6XXXX
            LLLLLLLLL  .........  '''''''''  ---------  =========  +++++++++  XXXXXXXXX
            LLLLLLLLL  .........  '''''''''  ---------  =========  +++++++++  XXXXXXXXX
           ==========================================================================
  FREQ.          2          1          1          0          1          1          0         10.00
        1    ILLLLLLI    I..1..I    I''2''I               I==4==I    I++5++I
        2    ILLLLLLI

TIME =     0.0
                                                          7          8          9          10
                                                    ========================================
                                                     OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
                                                     OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
                                                     OOOO7OOOO  ΘΘΘΘ8ΘΘΘΘ  ████9████  ████*████
                                                     OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
                                                     OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
                                                    ========================================
                                                          0          0          0          0
```

Example 1-10
(Continued)



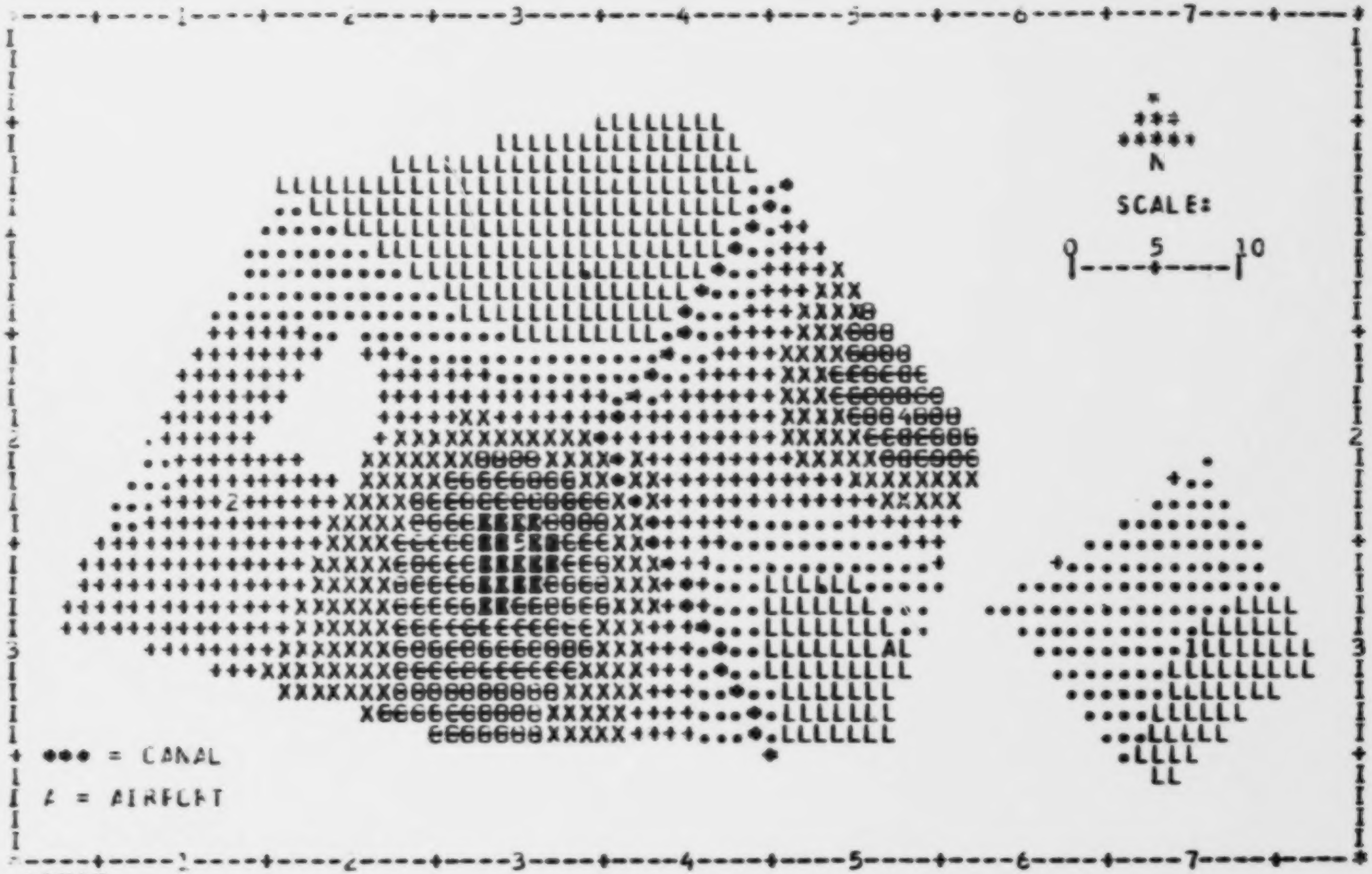
N
SCALE:
0
5
10
●●● = CANAL
A = AIRPORT
1
2
3
4
5
6
7

```
SYMAP
TIME =     0.0


TEST MAP #3
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
```

Example 1-11
Contour Map 3: Value Range Intervals

```
DATA VALUE EXTREMES ARE        -14.00       43.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
      ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)
                                                                              ABOVE
  MINIMUM      BELOW        0.0         5.00       10.00       20.00       30.00       40.00
  MAXIMUM       0.0         5.00       10.00       20.00       30.00       40.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

                           12.50       12.50       25.00       25.00       25.00

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL            L          1          2          3          4          5          H
          ===========================================================================
          LLLLLLLLL  .........  +++++++++  CCCCCCCCC  ΘΘΘΘΘΘΘΘΘ  ■■■■■■■■■  HHHHHHHHH
          LLLLLLLLL  .........  +++++++++  CCCCCCCCC  ΘΘΘΘΘΘΘΘΘ  ■■■■■■■■■  HHHHHHHHH
  SYMBOLS LLLLLLLLL  ....1....  ++++2++++  CCCC3CCCC  ΘΘΘΘ4ΘΘΘΘ  ■■■■5■■■■  HHHHHHHHH
          LLLLLLLLL  .........  +++++++++  CCCCCCCCC  ΘΘΘΘΘΘΘΘΘ  ■■■■■■■■■  HHHHHHHHH
          LLLLLLLLL  .........  +++++++++  CCCCCCCCC  ΘΘΘΘΘΘΘΘΘ  ■■■■■■■■■  HHHHHHHHH
          ===========================================================================
  FREQ.            2          1          0          1          0          1          1
TIME =      0.0
```

Example 1-11
(Continued)

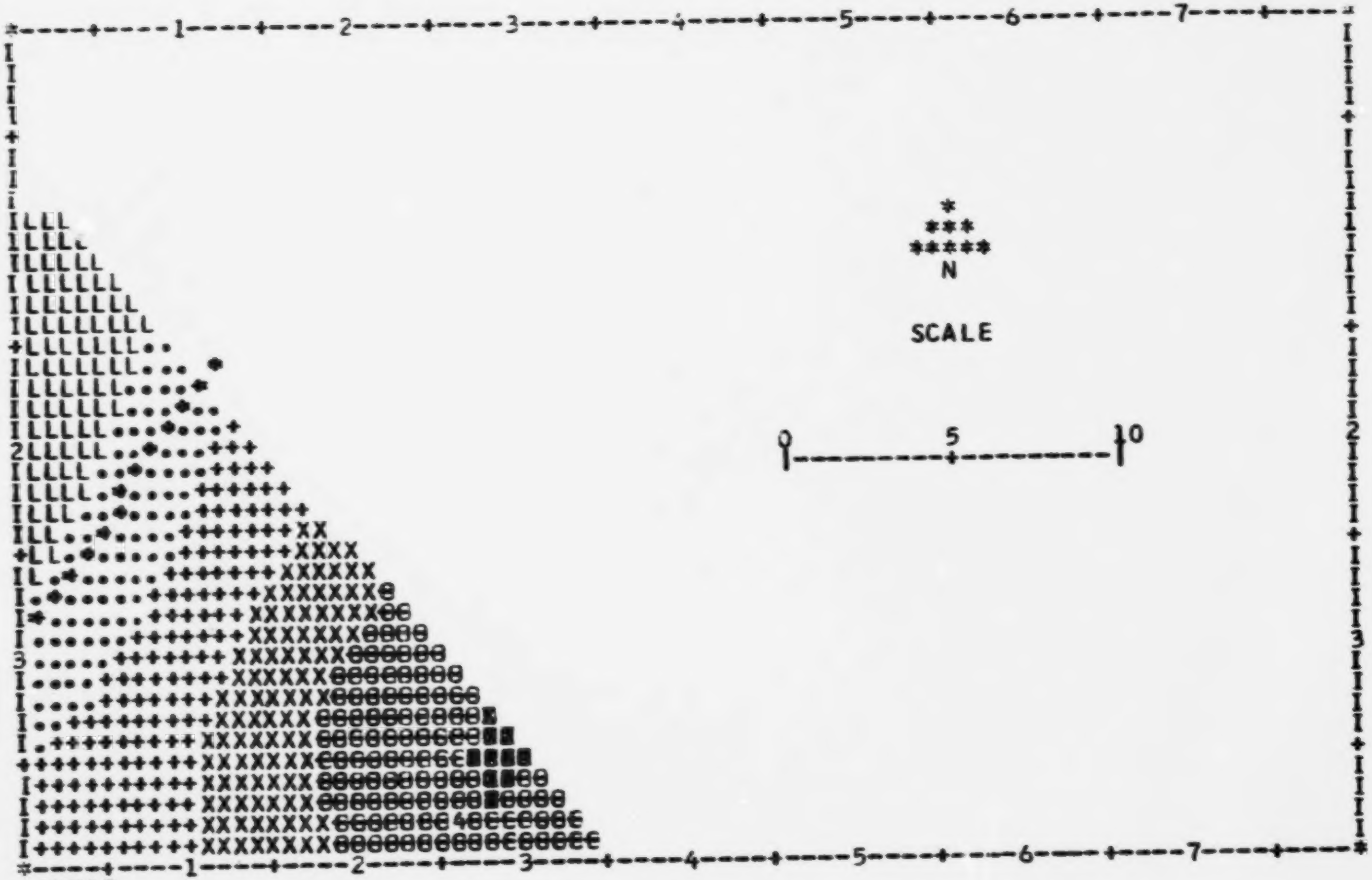

1
2
3
4
5
6
7
SCALE:
0
5
10
N
●●● = CANAL
A = AIRPORT

```
 SYMAP
TIME =     0.0

TEST MAP #6
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
MEASUREMENT IN INCHES
```

Example 1-12
Contour Map 4: Symbolism

```
DATA VALUE EXTREMES ARE        -14.00       43.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
        ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)

  MINIMUM      BELOW       0.0       10.00      20.00      30.00      40.00      50.00
  MAXIMUM       0.0       10.00      20.00      30.00      40.00      50.00     100.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

                          10.00      10.00      10.00      10.00      10.00      50.00

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL        L          1          2          3          4          5          6
           ==========================================================================
            LLLLLLLLL  .........  +++++++++  XXXXXXXXX  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
            LLLLLLLLL  .........  +++++++++  XXXXXXXXX. ΘΘΘΘΘΘΘΘΘ  █████████  █████████
  SYMBOLS   LLLLLLLLL  ....1....  ++++2++++  XXXX3XXXX. ΘΘΘΘ4ΘΘΘΘ  ████5████  ████6████
            LLLLLLLLL  .........  +++++++++  XXXXXXXXX  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
            LLLLLLLLL  .........  +++++++++  XXXXXXXXX  ΘΘΘΘΘΘΘΘΘ  █████████  █████████
           ==========================================================================
  FREQ.        2          1          1          0          1          1          0

TIME =     0.0
```

Example 1-12
(Continued)


N
SCALE
0
5
10
1
2
3
4
5
6
7

```
SYMAP

TIME =    0.0


TEST MAP #7
CONTOUR MAP - % CHANGE IN POPULATION FROM 1960 - 1970
```

Example 1-13
Contour Map 5: Inset Area

```
A-OUTLINE
                 1.0       1.6
                  .5       4.1
                 2.5       5.8
                 4.3       5.2
                 4.3       2.6
                 3.6        .2
                 1.0       1.6
                 3.5       5.8
                 2.6       7.1
                 3.8       7.8
                 4.6       6.8
                 3.5       5.8
99999
B-DATA POINTS
                 2.9       1.3
                 1.5       3.4
                 3.1       3.0
                 2.4       5.3
                 3.4       4.8
                 3.7       7.0
99999
C-OTOLEGENDS
    3    P        .6       6.8                -1.0
***
    5    P        .6       6.8       1.0      -2.0
*****
    1    P        .6       6.8      -1.0
*
    1    P        .6       6.8       2.0
N
    6    P        .6       6.8       5.0      -2.0
SCALE
         L       1.3       6.3
                 1.3       7.3
    1    P       1.3       6.3
1
    1    P       1.3       7.3
1
    1    P       1.3       6.8
•
    1    P       1.3       6.3      -1.0
0
    1    P       1.3       6.8      -1.0
5
    2    P       1.3       7.3      -1.0
10
   11    P       4.6        .2
```

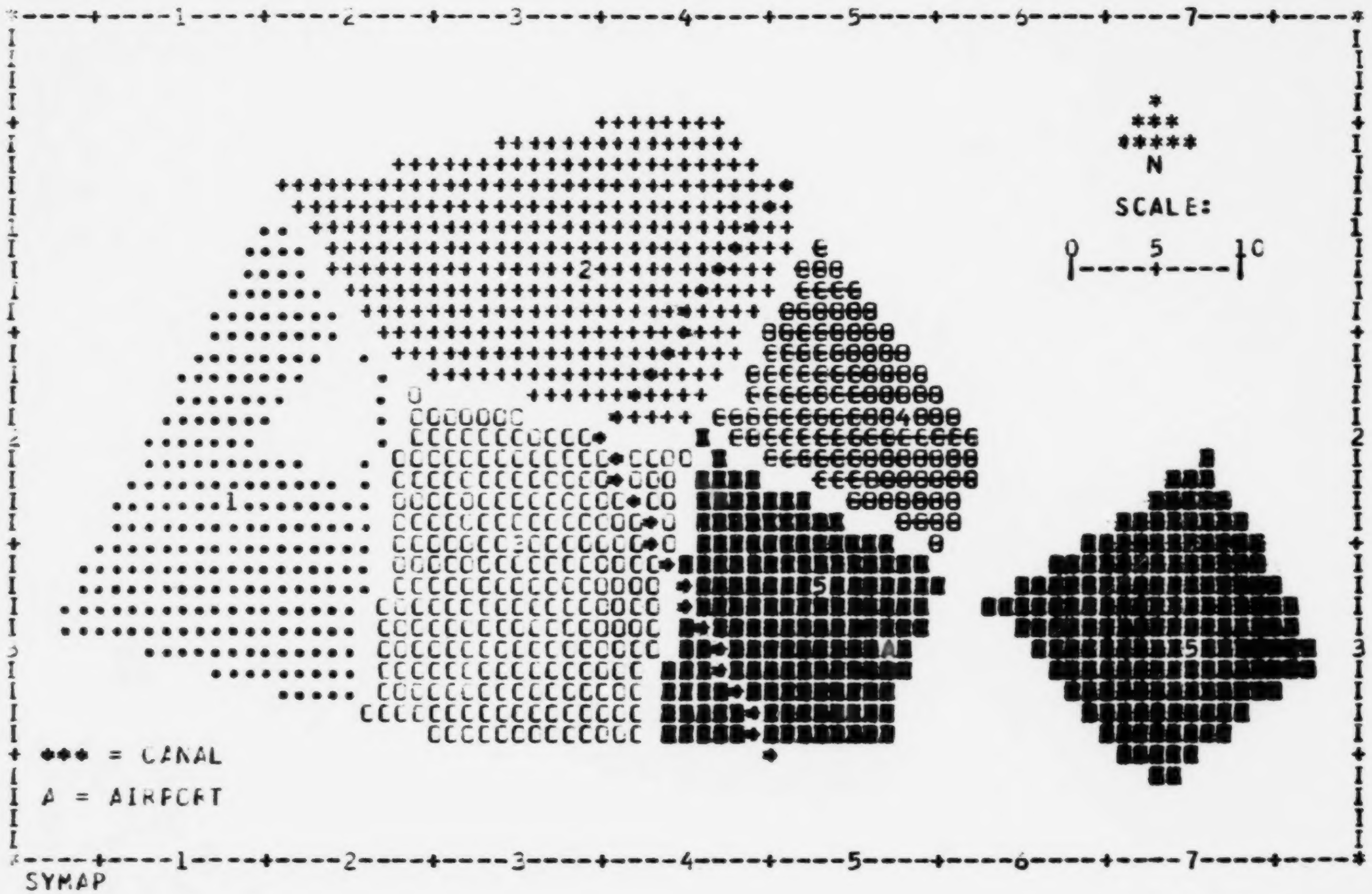
Example 1-14
Input Listing: Proximal Map

```
A = AIRPORT
        A           1.8         2.0
                    2.3         2.2
                    2.8         2.0
                    2.5         1.5
                    1.8         2.0
     A  P           3.8         5.2
    .=  L           4.4          .2
                    4.4          .4
    7   P           4.4          .6
= CANAL
    .=  L           1.0         4.6
                    2.5         3.5
                    4.4         4.5
99999
E-VALUES
                   10.0
                   20.0
                   30.0
                   40.0
                   50.0
                   60.0
99999
E-MAP
TEST MAP =3
PROXIMAL MAP
CLOSEST PROXIMITY TO TOWN HALL
    1               5.0         8.0
    2               0.0         0.0         5.0         8.0
   31
   36
   37
99999
C-OTOLEGENDS
99999
E-MAP
TEST MAP =8
PROXIMAL MAP
TOWN PRECINCTS
    1               5.0         8.0
    2               0.0         0.0         5.0         8.0
   31
   36
   37
    7
                               O
                               X
                               A
                               V
   11
99999
999999
```

Example 1-14
(Continued)

```
E-VALUES
--------


DATUM            VALUE

(    1)          10.00
(    2)          20.00
(    3)          30.00
(    4)          40.00
(    5)          50.00
(    6)          60.00



F-MAP
-----

TEST MAP #3

PROXIMAL MAP

CLOSEST PROXIMITY TO TOWN HALL

ELECTIVE

    1 MAP SIZE IS              5.00 INCHES LONG BY             8.00 INCHES WIDE
    2 EXTREME POINTS ARE (       0.0 ,       0.0 ) AND (        5.00,        8.00)
   31 FRACTIONAL EXTRAPOLATION IS         0.0
   36 PROXIMAL TYPE MAP
   37 VALUE COMPUTED INDEPENDENTLY FOR EVERY CHARACTER

TIME =     0.0

MAP     3
-------

TEST MAP #3
PROXIMAL MAP
CLOSEST PROXIMITY TO TOWN HALL




MAP SCALE =       1.0000 INCHES ON OUTPUT MAP/UNITS ON SOURCE MAP

MAP SHOULD BE PRINTED AT   8.0 ROWS PER INCH AND  10.0 COLUMNS PER INCH

ROW    = DOWN   COORDINATE *     8.0000
COLUMN = ACROSS COORDINATE *    10.0000


DATA POINTS FOR MAP

POINT          ROW    COLUMN      DATUM      VALUE      LEVEL

   1)           23       13          1      10.00          1
   2)           12       34          2      20.00          2
   3)           25       30          3      30.00          3
   4)           19       53          4      40.00          4
   5)           27       48          5      50.00          5
   6)           30       70          6      60.00          5
STANDARD SEARCH RADIUS IS            1.4127

TIME =     0.0
```

Example 1-15
Output Listing: Proximal Map

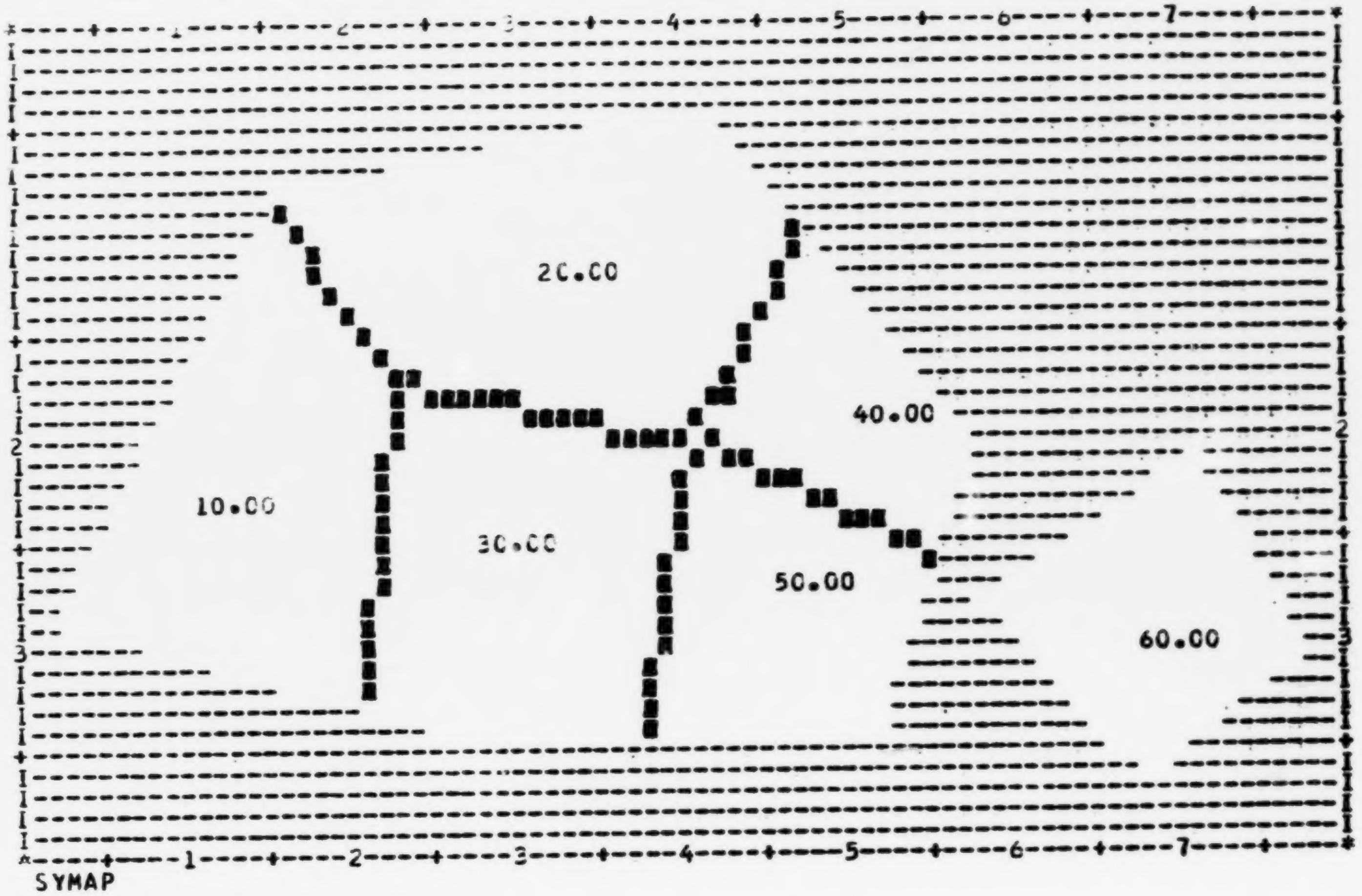

1
2
3
4
5
6
7
N
SCALE:
0
5
10
*** = CANAL
A = AIRPCRT

```
SYMAP
TIME =     0.0

TEST MAP #3
PROXIMAL MAP
CLOSEST PROXIMITY TO TOWN HALL
```

Example 1-16
Proximal Map 1: Standard Electives

```
DATA VALUE EXTREMES ARE        10.00       60.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
      ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)

  MINIMUM       10.00      20.00      30.00      40.00      50.00
  MAXIMUM       20.00      30.00      40.00      50.00      60.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

                20.00      20.00      20.00      20.00      20.00

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL           1          2          3          4          5
           ==================================================
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
  SYMBOLS   ....1....  ++++2++++  OOOO3OOOO  ΘΘΘΘ4ΘΘΘΘ  ████5████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
           ==================================================
  FREQ.           1          1          1          1          2
         :   I..1..I    I++2++I    IOO3OOI    IΘΘ4ΘΘI    I██5██I
         2                                              I██5██I

TIME =    0.0
```

Example 1-16
(Continued)



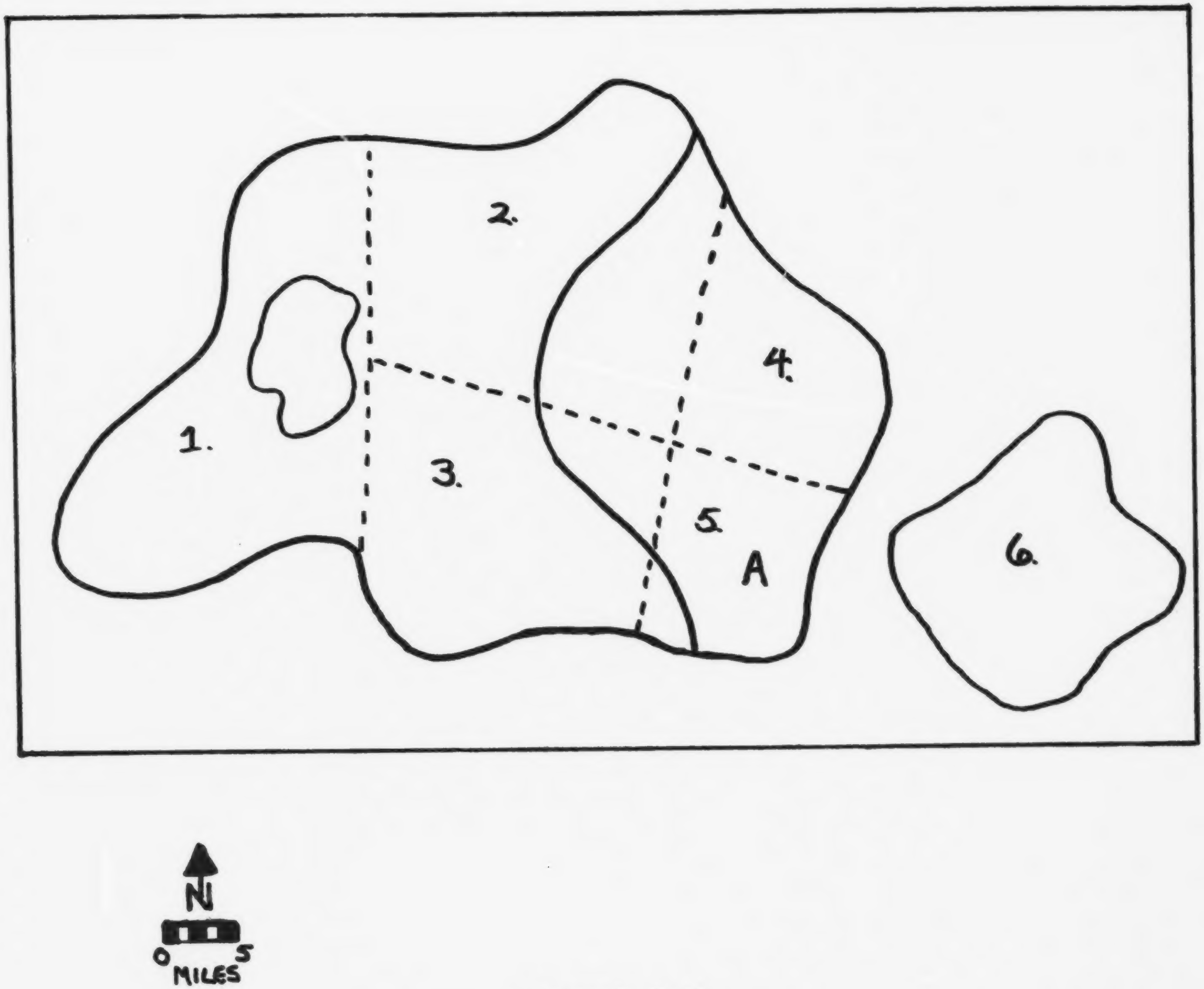
1
2
3
4
5
6
7
10.00
20.00
30.00
40.00
50.00
60.00

```
SYMAP

TIME =     0.0


TEST MAP #8

PROXIMAL MAP

TOWN PRECINCTS
```

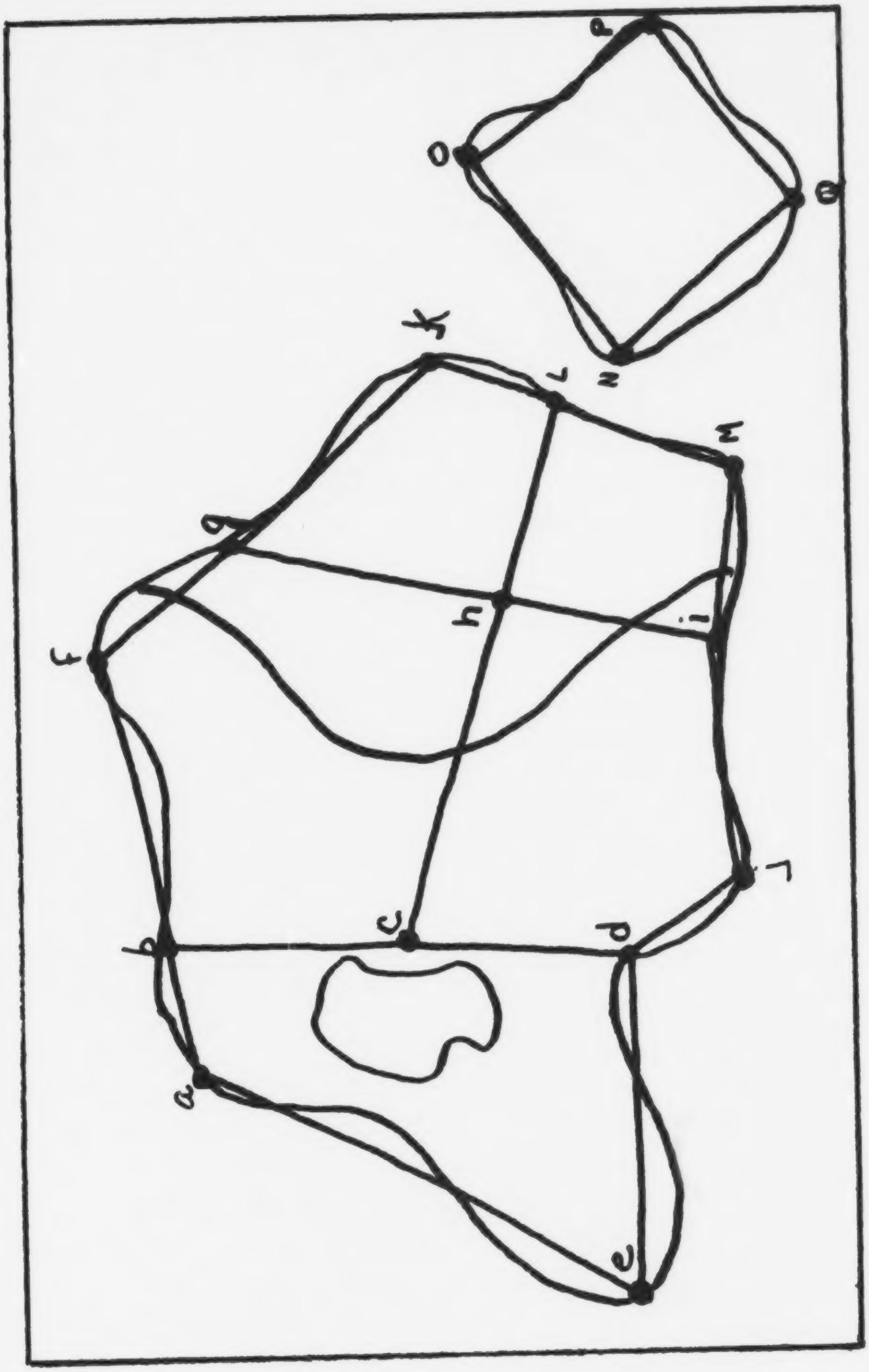
Example 1-17
Proximal Map 2: Special Symbolism

```
DATA VALUE EXTREMES ARE         10.00        60.00


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
        ('MAXIMUM' INCLUDED IN HIGHEST LEVEL ONLY)

  MINIMUM       10.00       20.00       30.00       40.00       50.00
  MAXIMUM       20.00       30.00       40.00       50.00       60.00

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

                20.00       20.00       20.00       20.00       20.00

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL           1           2           3           4           5
        ==========================================================

  SYMBOLS

        ==========================================================
  FREQ.           1           1           1           1           2
        1   I           I  I           I  I           I  I           I I           I
        2                                                            I             I
TIME =    0.0
```

Example 1-17
(Continued)


1.
2.
3.
4.
5.
A
6.
N
0
5
MILES

Example 1-18
Zone Source Map


a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q

Example 1-19
Zone Source Map With Conformant Boundaries

```
A-CONFORMCLINES
    1     A          1.0          1.7
                      .8          2.5
                     2.2          2.5
                     3.6          2.4
                     3.7           .3

                     1.0          1.7
    2     A           .8          2.5
                      .4          4.2
                     1.3          4.8
                     2.8          4.5
                     2.3          2.5
                      .8          2.5
    3     A          2.3          2.5
                     2.9          4.5
                     4.2          4.3
                     4.3          2.8
                     3.6          2.4
                     2.3          2.5
    4     A          1.3          4.8
                     2.5          6.0
                     3.2          5.7
                     2.9          4.5
                     1.3          4.8
    5     A          2.9          4.5
                     3.2          5.7
                     4.3          5.3
                     4.2          4.3
                     2.9          4.5
    6     A          3.7          6.0
                     2.8          7.2
                     3.8          8.0
                     4.7          6.9
                     3.7          6.0
99999
```

Example 1-20
Input Listing: Conformant Map

```
C-OTOLEGENDS
    3    P        .6       6.8                  -1.0
***
    5    P        .6       6.8        1.0       -2.0
*****
    1    P        .6       6.8       -1.0
*
    1    P        .6       6.8        2.0
N
    6    P        .6       6.8        4.0       -2.0
SCALE:
    1    P        .6       6.8        6.0
5
   11    P        .6       6.8        7.0       -5.0
|----+----|
    1    P        .6       6.8        6.0       -5.0
0
    2    P        .6       6.8        6.0        5.0
10
   11    P       4.6        .2
A = AIRPORT
         A       1.8       2.0
                 2.3       2.2
                 2.8       2.0
                 2.5       1.5
                 1.8       2.0
     A   P       3.8       5.2
     +=  L       4.4        .2
                 4.4        .4
    7    P       4.4        .6
= CANAL
     +=  L       1.0       4.6
                 2.5       3.5
                 4.4       4.5
99999
E-VALUES
                40.0
                12.0
               100.0
                30.0
                96.0
                73.0
99999
F-MAP
TEST MAP #4
CONFORMANT MAP - TAX RATE

    1            5.0       8.0
    2            0.0       0.0        5.0        8.0
99999
```

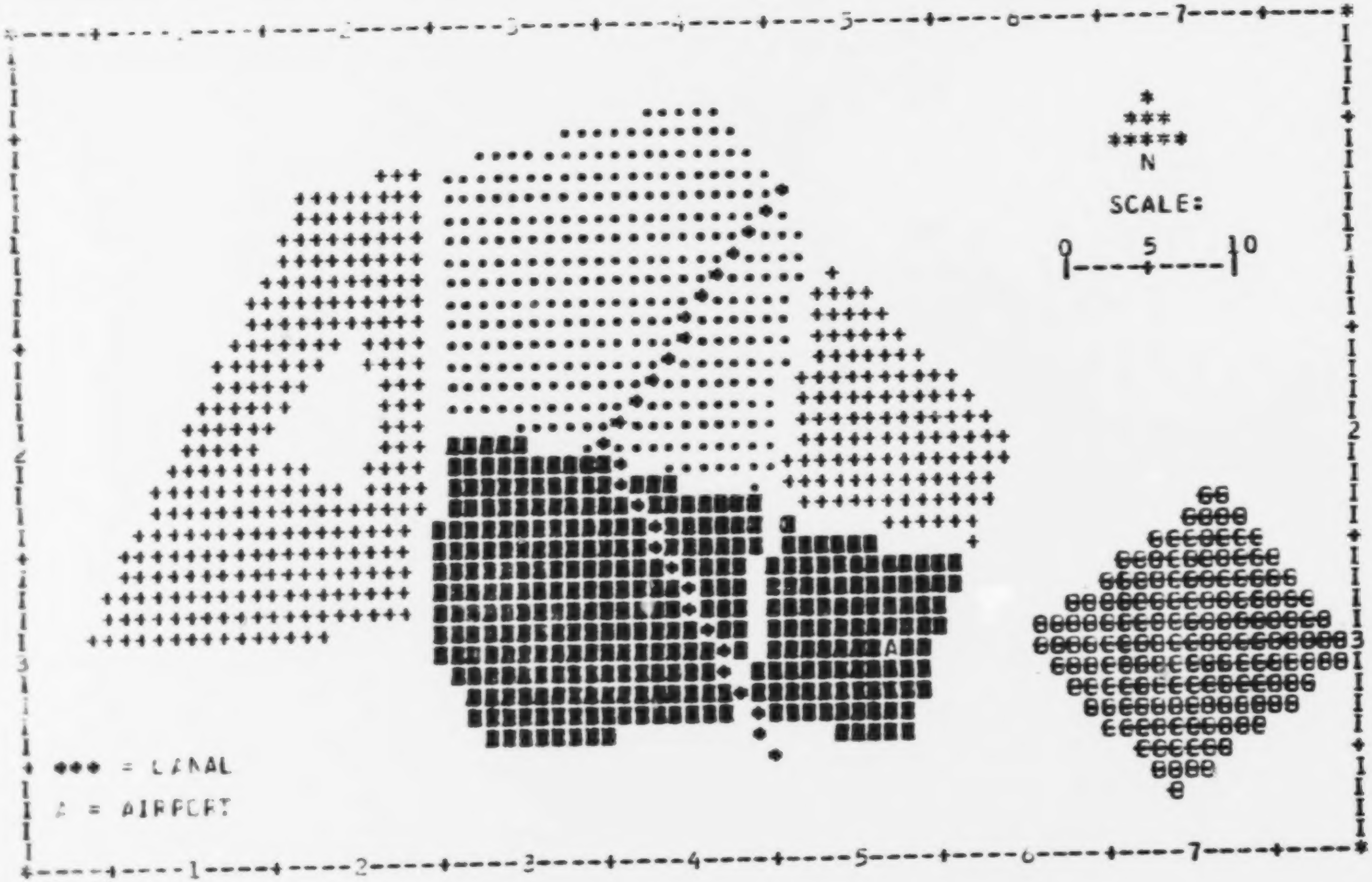
Example 1-20
(Continued)

```
A-CONFORMCLINES
---------------

VERTEX              DOWN      ACROSS
     1     AREA

(    1)              1.00      1.70
(    2)              0.80      2.50
(    3)              2.20      2.50
(    4)              3.60      2.40
(    5)              3.70      0.30
(    6)              1.00      1.70
LENGTH=         8.77
AREA=        3.94
CENTER=(        2.48,         1.69)
     2     AREA

(    1)              0.80      2.50
(    2)              0.40      4.20
(    3)              1.30      4.80
(    4)              2.80      4.50
(    5)              2.30      2.50
(    6)              0.80      2.50
LENGTH=         7.92
AREA=        4.18
CENTER=(        1.58,         3.64)
     3     AREA

(    1)              2.30      2.50
(    2)              2.90      4.50
(    3)              4.20      4.30
(    4)              4.30      2.80
(    5)              3.60      2.40
(    6)              2.30      2.50
LENGTH=         7.02
AREA=        3.17
CENTER=(        3.38,         3.38)
     4     AREA

(    1)              1.30      4.80
(    2)              2.50      6.00
(    3)              3.20      5.70
(    4)              2.90      4.50
(    5)              1.30      4.80
LENGTH=         5.32
AREA=        1.61
CENTER=(        2.42,         5.19)
     5     AREA

(    1)              2.90      4.50
(    2)              3.20      5.70
(    3)              4.30      5.30
(    4)              4.20      4.30
(    5)              2.90      4.50
LENGTH=         4.73
AREA=        1.38
CENTER=(        3.63,         4.94)
     6     AREA

(    1)              3.70      6.00
(    2)              2.80      7.20
(    3)              3.80      8.00
(    4)              4.70      6.90
(    5)              3.70      6.00
LENGTH=         5.55
AREA=        1.91
CENTER=(        3.75,         7.02)
```

Example 1-21
Output Listing: Conformant Map

```
E-VALUES
--------


DATUM             VALUE

(    1)            40.00
(    2)            12.00
(    3)           100.00
(    4)            30.00
(    5)            56.00
(    6)            73.00


F-MAP
-----

TEST MAP #4

CONFORMANT MAP - TAX RATE


ELECTIVE

    1 MAP SIZE IS              5.00 INCHES LONG BY             8.00 INCHES WIDE
    2 EXTREME POINTS ARE (        0.0 ,        0.0 ) AND (        5.00,        8.00)
TIME =     0.0


MAP   1
-------

TEST MAP #4
CONFORMANT MAP - TAX RATE




MAP SCALE =     1.0000 INCHES ON OUTPUT MAP/UNITS ON SOURCE MAP

MAP SHOULD BE PRINTED AT   8.0 ROWS PER INCH AND  10.0 COLUMNS PER INCH

ROW    = DOWN   COORDINATE *     8.0000
COLUMN = ACROSS COORDINATE *    10.0000


CONFORMLINES FOR MAP
 ZONE         TYPE         ROW      COLUMN       DATUM      VALUE       LEVEL

     1)         A           20         17           1        40.00         2
     2)         A           13         36           2        12.00         1
     3)         A           27         34           3       100.00         5
     4)         A           19         52           4        30.00         2
     5)         A           29         49           5        56.00         5
     6)         A           30         70           6        73.00         4
 TIME =    0.0
```

Example 1-21
(Continued)



*
***
*****
N
SCALE:
0 5 10
*** = CANAL
A = AIRPORT
1
2
3
4
5
6
7

```
SYMAP
TIME =     0.0

TEST MAP 64
CONFORMANT MAP - TAX RATE
```

Example 1-22
Conformant Map

```
DATA VALUE EXTREMES ARE        12.CC      10C.CC


ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
       ('MAXIMUM' INCLUDED IN HIGHEST LEVEL CNLY)

  MINIMUM      12.00       29.60       47.20       64.80       82.4C
  MAXIMUM      29.60       47.2C       64.80       82.40      100.CC


PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

               20.CC       20.00       20.CC       2C.00       2C.00


FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL
  LEVEL           1          2          3          4          5
           ================================================================
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
  SYMBOLS   ....1....  ++++2++++  OOOO3OOOO  ΘΘΘΘ4ΘΘΘΘ  ████5████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
            .........  +++++++++  OOOOOOOOO  ΘΘΘΘΘΘΘΘΘ  █████████
           ================================================================
  FREQ.           1          2          0          1          2
          1   I..1..I    I++2++I               IΘΘ4ΘΘI    I██5██I
          2              I++2++I                          I██5██I

TIME =    0.0
```

Example 1-22
(Continued)

2. PREPARATION OF PACKAGES

To produce computer maps, a deck of punched cards must be prepared as input to the computer. This deck will consist of certain introductory cards, a copy of the SYMAP program on cards (or on tape), and a number of "packages," each composed of additional cards covering a specific category of information about the map to be produced. The available packages are:

i) A-OUTLINE--used to describe the outline of the study area, by specifying the coordinate locations of the outline vertices. (Used for contour and proximal maps only.)

ii) A-CONFORMOLINES--used to give the position of the zones to which data is to be related, by specifying the coordinate locations of vertices on the zonal outlines. This package is required for a conformant map.

iii) B-DATA POINTS--used to give the position of the points to which your data is to be related, by specifying their coordinate locations. Data points may be either the points for which data is available, or the centers of areas, called data zones, for which data is available. This package is required for contour and proximal maps.

iv) C-OTOLEGENDS--used to cause supplementary information to appear on the face of the map by specifying their coordinate locations and content. The relative positions of legends are adjusted automatically if the size and/or scale of the map are altered.

v) D-BARRIERS--used to give the coordinate location and strength of impediments to interpolation at specified vertices.

vi) E-VALUES--used to assign numerical data to the data points and/or data zones, by specifying the "values" involved.

vii) E1-VALUES INDEX--used to adjust the reference order of data values in the E-VALUES package.

viii) F-MAP--used to instruct the computer to make each map according to certain "electives," most of which are optional and have a standard default. It is also used to specify below the map an appropriate title for each map. An F-MAP package is required for each map desired.

2.1 DETERMINING LOCATIONS ON A SOURCE MAP

The geographic area to be mapped is referred to as the "study area." Normally, a "source map" will be used with designated top and left map borders from which required measurements may be taken. This source map may be identical in size to the maps which are to be produced or it may be of any size, with adjustment in size and scale obtained automatically through the use of certain specified electives in the F-MAP package.

A location on the source map is expressed in terms of two coordinates, the first measured vertically from the top map border, the second measured horizontally from the left map border. There are two different systems for measuring coordinate distances; row and column measurement and equal unit measurement.

2.1.1 Row and Column Measurement

Standard computer printers, properly set to print 8 rows per inch, produce output with characters located in rows one-eighth inch in height and in columns one-tenth inch in width. The standard output "grid" for mapping is thus composed of "character locations" or cells each measuring one-eighth inch vertically by one-tenth inch horizontally. Coordinate measurements may be figured in terms of these cells (or their centers). A special SYMAP ruler (and plastic grid sheet), combining both units, is available to facilitate the measurement of coordinate locations on the source map in conformance with their position on the maps to be produced.

The characters (-'s and I's) which form the rectangular borders on the maps produced by SYMAP are centered in the rows and columns which they respectively occupy. The top border centers in what is considered to be row zero. All other rows follow vertically down the length of the map. Correspondingly, the left border centers in what is considered to be column zero. All other columns follow horizontally across the width of the map.

The location of a point is specified by the number of the row (down from the top border), followed by the number of the column (across from the left border) in which the point lies. Locations should be adjusted as required to conform to the grid. If a location falls precisely between grid cells, shift it to the cell on the right or to the cell below, as the case may be.

2.1.2 Equal Unit Measurement

Although the output map is printed in the rows and columns of the computer grid, the SYMAP program is written to accept measurements in any units of equal size as standard. The same scale of measurement is used to determine both vertical and horizontal coordinates. For example, using the inch scale on a map ten inches square, the location of a point half way down the map and a quarter of the way across the map from the left is specified as "5.0" (vertical coordinate) followed by "2.5" (horizontal coordinate). The scale selected must be used consistently in determining all locations for any one map (or for a series of related maps) except where the instructions state otherwise.

Since the SYMAP program is written to accept equal unit measurement as standard, when using row and column measurement, the number "8." and the number "10." must be punched on the first card of the A-OUTLINE, A-CONFORMOLINES, B-DATA POINTS, C-OTOLEGENDS, and D-BARRIERS packages, if they are included. These numbers instruct the computer to convert the specified row and column measurements to equal unit measurements.

2.1.3 Use of "Legends," "Title," and "Text"

Wording, numbers, or symbols common to all maps of a series should ordinarily be treated as legends on the face of the map--not as title or text. So treated they will appear automatically on all maps of a series. (See C-OTOLEGENDS packages.)

The title for each map should be clear and meaningful--differentiated from all other maps in the same series. It should provide information basic to the viewer's general comprehension of what is being presented. The text should provide significant supplementary information. (See F-MAP package.)

2.2 A-OUTLINE

This package is optional and is used to specify the outline of the study area for a contour or a proximal map, when the study area does not fill the entire space within the rectangular map border.

On the first card of this package, punch "A-OUTLINE" in columns 1-9, and the letter "X" in column 23 if tabular printout of the input data is not desired. (If coordinate measurements are by row and column, continue punching the first card with the number "8." in columns 31-32 and the number "10." in columns 41-43.)

On the last card, punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the coordinate locations of the study area outline vertices (namely, those points at which the outline changes its direction). Punch each vertex location on a separate card, starting with the uppermost vertex and proceeding clockwise back to and including once again the point of beginning. This repetition tells the computer that the outline is complete. If there are two or more vertices equally high, start with the one furthest to the left. If the outline is curved, approximate the curve with short straight-line segments meeting at what will then be vertices. If the outline is overly complex, it may be simplified as desired. An approximate outline will usually prove adequate.

Punch the vertical coordinate (measurement down from the top border) as a decimal number right justified in columns 11-20. Punch the horizontal coordinate (measurement across from the left border) as a decimal number right justified in columns 21-30.

If the study area is not contained within a single outline, two or more outlines may be employed--presented in any desired sequence. There is no set limitation on the possible number of outlines, but no one outline in this package may have fewer than three nor more than one hundred vertices. If a large or complex outline would require more than one hundred vertices, subdivide it into two or more outlines which meet along a common and nonhorizontal edge.

2.3 A-CONFORMOLINES

The A-CONFORMOLINES package--short for conformant outlines--is used to specify the outline of each of the data zones of the study area. Only one data value may be associated with any one data zone. In certain instances, however, more than one outline may be needed to define a data zone, as for the Hawaiian Islands. In such cases, each of the outlines which together define the whole data zone is associated with the same data value.

The outline of a zonal area is determined on the source map and specified on cards by a procedure somewhat similar to that employed in the A-OUTLINE package (not used for conformant mapping).

The outline of each data zone should be designated on the source map. If the outline is curved, approximate the curve with straight-line segments. If the outline is overly complex, it may be simplified as desired; an approximate outline will usually prove adequate. The outline of a data zone will most commonly be designated by at least three points. However, a data zone may be so relatively small that it must be designated by a single point, or so narrow that it must be designated by two or more points as a line.

On the first card of this package, punch "A-CONFORMOLINES" in columns 1-15, and the letter "X" in column 23 if tabular printout of the input data is not desired. (If coordinate measurements are by row and column, continue punching the first card with the number "8." in columns 31-32 and the number "10." in columns 41-43.) Punch "PU" in columns 26-27 if you want the calculated centers of gravity of the conformant zones to be punched on cards using the format of the B-DATA POINTS. Punch "PR" in columns 29-30 if you want these zone centers printed as data points on the output map. Punch the letter "X" in column 63 if input is on tape (unit 11). If the A-CONFORMOLINES package is on tape, the entire package (including the 99999 card) following the package name card must be on tape. The tape format is the same as the standard form for this package. The tape is rewound after the 99999 is read.

On the last card, punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the specifications for each of the zonal outlines, completing all of the cards for the first outline before going on to the second, etc.

Extreme care must be taken to ensure that each data zone is correctly associated with its corresponding data value, and on the first card for each conformant outline, columns 1-5 are available for punching the reference number.

of the associated data value. However, these columns may be left blank if zonal outlines are entered in precisely the order in which their associated data values are listed, and when only one outline is included for each zone. Sequential reference to the values as listed in the E-VALUES package is automatic--the first outline refers to the first value, the second to the second, and so on.

On the first card for each conformant outline, punch (or leave blank) the reference number of the associated data value as an integer right justified in columns 1-5. In column 10, punch the letter "A" or "L" or "P" to indicate whether the zonal outline is to be represented as an area, a line, or a point. Columns 11-20 and 21-30 are used to specify the coordinate location of the first vertex or point of the zonal outline. For an "area" this will be the location of the first vertex on its circumference; for a "line," its starting point; for a "point," simply the location of that point. This first card is the only one required for a "point" representation. Punch the vertical coordinate (measurement down from the top map border) as a decimal number right justified in columns 11-20. Punch the horizontal coordinate (measurement across from the left map border) as a decimal number right justified in columns 21-30.

On additional cards, the coordinate locations of the remaining vertices or points are specified in the order of their position on the outline, one location per card, punched as on the first card. For an "area" outline, the location of the first vertex should be repeated to "close" the outline. Nothing is punched in columns 1-10 on these cards.

2.4 B-DATA POINTS

This package is used to specify the coordinate locations of the points at which data is to be provided. Data points may be located outside the study area, and even beyond the rectangular map border. In the latter event, however, their location will not appear. No special sequence of location is required, but if the locations or data to be used are already in some established order, it will usually be convenient to use that sequence. If a conformant map is produced from this source map, the reference number of each data point should be the same as that of the zonal outline in which it appears for purposes of checking.

On the first card of this package, punch "B-DATA POINTS" in columns 1-13, and the letter "X" in column 23 if tabular printout of the input data is not desired. If coordinate measurements are by row and column, continue punching the first card with the number "8." in columns 31-32 and the number "10." in columns 41-43.

On the last card, punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the coordinate locations of the data points, each data point on a separate card. There is a limit of 1000 data points for any one map. If more data points are needed, divide the work into two or more parts with some overlap (or increase the dimensions in the SYMAP program deck for the appropriate variables--see Section 4). The sequence of the cards in this and the corresponding E-VALUES package is very important. Punch the vertical coordinate (measurement down from the top border) as a decimal number right justified in columns 11-20. Punch the horizontal coordinate (measurement across from the left border) as a decimal number right justified in columns 21-30. The coordinate for a data point falling on the top (or left) border should be "0." (zero). Data points beyond these two borders would be located by means of negative numbers--preceded by a minus sign.

2.5 C-OTOLEGENDS

This package is used to specify the relative position and content of any special wording, numbering or other symbolism desired on the face of the map or within the rectangular map border. Any supplementary information which will apply equally to all the maps in any one series may be provided, such as: the general title applicable to the study area, compass directions, major landmarks, rivers and railroads. As maps may be run at different scales, it is recommended that the scale of the map be shown by a graphic scale, without reference to inches or other fixed dimensional units.

The map background--the area between the rectangular map border and the outline of the study area--

may be used for legends to an extent deemed suitable by the user. Within the study area legends may adversely affect map legibility and comprehension, especially if placed at data point locations. There is no set limit on the number of legends.

The C-OTOLEGENDS package enables the user to specify legends as strings of characters, and symbols, to be placed on the output map in relation to the vertical and horizontal coordinates of the source map. The actual grid locations of these legends on the output map will depend on the use made of electives 1, 2, 13, 14 and 15 in establishing the size and scale of the map. Coordinate locations are determined either by equal unit, row and column measurement. All coordinate locations are specified by decimal numbers right justified in the columns to which they are assigned.

On the first card of this package, punch "C-OTOLEGENDS" in columns 1-12, and the letter "X" in column 23 if printout of the input data is not desired. (If coordinate measurements are by row and column, continue punching the first card with the number "8." in columns 31-32 and the number "10." in columns 41-43.)

On the last card, punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the specifications for the legends desired, as shown in the accompanying Table 2-1. The card s

TABLE 2-1

SPECIFICATIONS FOR OTOLEGENDS

(i) POINT LEGEND, SINGLE SYMBOL--Overprinted, if desired--1 card

Columns 6-9	The print and overprint characters (any of which may be blank) for the single symbol desired.
Column 10	The letter "P".
Columns 11-20	The vertical coordinate of associated source map point.
Columns 21-30	The horizontal coordinate of associated source map point.
Columns 31-40	The vertical displacement desired, namely, the number of rows up (precede by "-"), or the number of rows down for the symbol to be adjusted, relative to its associated source map point.
Columns 41-50	The horizontal displacement, namely, the number of columns to the left (precede by "-"), or the number of columns to the right for the symbol to be adjusted, relative to its associated source map point.

(ii) POINT LEGEND, MULTIPLE CHARACTER (Vertical or Horizontal)--No overprint--2 cards

First Card:

Column 1	Leave blank for horizontal legend, punch "-" (minus) for vertical legend.
Columns 4-5	The number of characters in legend (not to exceed 50).
Column 10	The letter "P".
Columns 11-20	The vertical coordinate of associated source map point.

TABLE 2-1, cont'd.

Columns 21-30	The horizontal coordinate of associated source map point.
Columns 31-40	The vertical displacement, namely, the number of rows down for the "start" of the legend, relative to its associated source map point.
Columns 41-50	The horizontal displacement, namely, the number of columns to the left (precede by "-"), or the number of columns to the right for the "start" of the legend, relative to its associated source map point.

Second Card:

Columns 1-50	Punch the desired legend starting in Column 1 and ending in the column whose number is punched in Columns 4-5 of the first card.

(iii) LINE LEGEND, SINGLE SYMBOL--Repeated--2 or more cards

First Card:

Columns 6-9	The print and overprint characters (any of which may be blank) for the symbol desired.
Column 10	The letter "L".
Columns 11-20	The vertical coordinate of first point on line.
Columns 21-30	The horizontal coordinate of first point on line.

Other Cards:

The coordinate locations of the succeeding vertices on the line, one location to a card, in columns 11-20 and 21-30 as for the first point. Columns 1-10 are left blank on these cards.

TABLE 2-1, cont'd.

(iv) AREA LEGEND, SINGLE SYMBOL--Filled outline--2 or more cards

First Card:

Columns 6-9	The print and overprint characters (any of which may be blank).
Column 10	The letter "A".
Columns 11-20	The vertical coordinate of the first vertex (the uppermost point on the outline, and if more than one, the left most of these).
Columns 21-30	The horizontal coordinate of the first vertex.

Other Cards:

The coordinate locations of succeeding vertices on the outline, one location to a card, in columns 11-20 and 21-30 as for the first vertex. On the last card repeat the coordinate location of the first vertex to "close" the outline. Columns 1-10 are left blank on these cards.

required for each legend must, of course, be in correct sequence, but the OTOLEGENDS may be specified in any order regardless of type or location on the map.

2.5.1 Characters and Symbols

A character is any one of the following key punch designations: 0123456789ABCDEFGHIJKLMNOPQRSTUV WXYZ+.)-$*/,(=' and the blank. Additional characters are available on some newer computer machinery.

A symbol is composed of four characters, printed one on top of the other in the same location, any or all of which may be blank. This process is called over-printing. Examples of such symbols are "Ø", "⊠", "■". The last symbol is composed of the characters "O", "X", "A", "V", and usually appears as solid black on most printers.

A string is an ordered set of one or more characters; e.g., "NEW YORK" or "*/*/*/".

2.5.2 Types of Legends

There are three major types of legends which can be specified in the C-OTOLEGENDS package. Each type is associated with a distinct type of figure on the source map, as follows:

2.5.2.1 Point Legend

This type of legend is associated with a given point on the source map and can be either (i) a horizontal row of characters, (ii) a vertical row of characters--a row of characters may not include overprints--(iii) or a single symbol--composed of four characters overprinted, any or all of which may be blank.

Horizontal and vertical OTOLEGENDS require two cards for each legend. On the first card, column 1 is left blank if the character string is to proceed horizontally across, or contains a minus sign ("-") if the character string is to proceed vertically down the map. Columns 2 and 3 are blank, and the number of characters in the string is punched as an integer right justified in columns 4-5. This number may not exceed 50. Columns 6-9 are blank, and the letter "P" is punched in column 10 to indicate that the associated figure is a point. The location of the point legend on the source map is specified by punching the vertical coordinate in columns 11-20, and the horizontal coordinate in columns 21-30.

If the first character of the string is not to appear at the row and column location of the associated source map point, punch as a decimal number right justified in columns 31-40 the number of rows to be added or subtracted and/or punch the number of columns to be added or subtracted in columns 41-50, as shown in Example 2-1 which follows.

On the second card the character string desired is punched, beginning in column 1. It may not extend beyond column 50 and should end in the column whose number appears in columns 4 and 5 of the preceding card.

EXAMPLE 2-1

POINT LEGEND

Assume that we wish to indicate the location of Akron, Ohio on a map with the character string "AKRON", and that the vertical and horizontal coordinates of Akron on the source map are 7.43 and 2.31. If we leave card columns 31-50 blank, the letter "A" of "AKRON" will be printed (on the output map) at the associated location of the point with coordinates 7.43 and 2.31. If we wish to center the name "AKRON" so that the "R" would fall on the precise location of the city, the legend would be moved two columns to the left by specifying "-2.0" in columns 47-50. The two cards for the latter legend would be punched:

columns	1-5	10	11-20	21-40	31-40	41-50
Card 1:	5	P	7.43	2.31		-2.0
Card 2:	AKRON					

If, in addition, we wished the name "AKRON" to be centered and to appear on the line below the actual location of the city, "1.0" would be punched in columns 38-40 as the number of rows to be added. (We might wish to do this if a data point symbol is to appear on the actual location.) The two cards for this legend would be punched:

Card 1:	5	P	7.43	2.31	1.0	-2.0
Card 2:	AKRON					

A single symbol legend requires only one card and is specified as follows: Leave the first five columns blank; punch the print and overprint characters for the desired symbol in columns 6-9 (any of which may be left blank), punch the letter "P" for POINT in column 10; and punch the vertical coordinate of the point in columns 11-20, and the horizontal coordinate in columns 21-30.

If the user does not wish to have the symbol appear exactly on top of the associated point, but rather a number of rows or columns away from this point, he may specify this in columns 31-40 and 41-50 as indicated above.

2.5.2.2 Line Legend

Line legends resemble single symbol point legends--the symbol appears everywhere along the specified line rather than at a single point. Line legends require two or more cards and are specified as follows:

Columns 1-5 are left blank; punch the print and overprint characters for the desired symbol in columns 6-9 and the letter "L" for LINE in column 10; then punch the vertical coordinate of the first point on the line in columns 11-20 and the horizontal coordinate in columns 21-30. Nothing is punched in columns 31-50 since there is no capability for row or column displacement with LINE and AREA legends. Additional vertices on the line (points at which the line changes direction) are punched--one location to a card--on succeeding cards in columns 11-20 and 21-30 as for the first point. Columns 1-10 are left blank on these cards, as shown in Example 2-2.

2.5.2.3 Area Legend

An area legend causes the entire area within an outline--similar to the A-OUTLINES package--to be filled with the specified symbol. Area legends require three or more cards and are specified as follows: Columns 1-5 are left blank; punch the print and overprint characters for the desired symbol in columns 6-9 and the letter "A" for AREA in column 10. Then punch the vertical coordinate of the first vertex on the outline of the area in columns 11-20 and the horizontal coordinate in columns 21-30.

Additional vertices are punched on succeeding cards in columns 11-20 and 21-30 as for the first vertex, with one coordinate location per card. Repeat the coordinate location of the first vertex to "close" the outline, as shown in Example 2-3.

2.6 D-BARRIERS

This package is used to specify any desired barrier to interpolation between data points. A barrier may be used to modify the computer's distance-oriented interpolation in order to reflect the probable effect of an obstacle, such as a body of water, a boundary or government control, or an expressway. For example, on a population density map of the Boston-Cambridge area, the value computed at a point

EXAMPLE 2-2

LINE LEGEND

To indicate a railroad running from Buffalo to Albany and then to New York City, we might use a line legend filled with the symbol "R̸". Three cards would be punched as follows:

Columns	6-9	10	11-20	21-30
Card 1:	R/	L	2.31	4.08
Card 2:			2.35	7.92
Card 3:			5.81	8.01

The location of Buffalo is on the first card, Albany on the second and New York City on the third. Note that the "R" and "/" would not be printed side by side, but on top of one another

EXAMPLE 2-3

AREA LEGEND

Using a modification of Example 2-2, for illustration:

Card 1:	R/	A	2.31	4.08
Card 2:			2.35	7.92
Card 3:			5.81	8.01
Card 4:			2.31	4.08

would cause the entire triangle formed by Buffalo, Albany, and New York City to be filled in with the symbol "R̸".

near the Charles River would normally depend on--and reflect--the data from surrounding tracts on both sides of the river. But the river probably forms a physical, psychological, and political barrier to the distribution of population, the effect of which the user may want included. He therefore imposes a restriction upon the interpolation across the river through the use of a barrier--which may be either permeable or impermeable. Impermeable barriers allow no interpolation to occur across the barrier; permeable barriers restrict interpolation across the barrier but do not stop it.

Because barriers increase the cost and complexity of maps, they should be provided only where their omission would result in obviously invalid interpolation. Barriers are not used for conformant or Proximal maps since no interpolation takes place.

On the first card of this package, punch "D-BARRIERS" in columns 1-10, and the letter "X" in column 23 if tabular printout of the input data is not desired. (If coordinate measurements are by row and column, continue punching the first card with the number "8." in columns 31-32, and the number "10." in columns 41-43.)

On the last card punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the coordinate locations of the barrier vertices--points at which the barrier may begin, change direction, and end--followed by the strength desired.

If a barrier has one end within the map and the other at the border, make the first vertex the interior end and work toward the border. For other types of barriers, begin with the uppermost terminal vertex. If both terminal vertices are equally high, start with the one furthest to the left. Punch vertices in the order in which they occur along the barrier, one vertex per card; proceed clockwise if the barrier is closed. Always repeat the last vertex of a barrier. This repetition tells the computer that the barrier is complete. Position barriers so that they would intercept all imaginary lines connecting character locations and data points between which interpolation is to be impeded by the barrier.

Punch the vertical coordinate of the vertex as a decimal number right justified in columns 11-20, and the horizontal coordinate as a decimal number right justified in columns 21-30. Punch the strength as a decimal number right justified in columns 31-40. Permeable barriers may assume different strengths at succeeding vertices, but they cannot become impermeable. If the barrier is impermeable--no interpolation whatever is to occur across it--punch a strength of "-1." in columns 38-40.

Two or more barriers may be employed--entered in any desired sequence. Each barrier acts independently, and they may cross each other at vertices or elsewhere, or coincide for any portion of their length. There is no limitation on the possible number of barriers, provided that the total number of vertices does not exceed 50--excluding repetitions of terminal vertices.

2.7 E-VALUES

This package is used to specify the values or quantitative information applicable to each data point (for a contour or proximal map) or to each data zone (for a conformant map).

On the first card of this package, punch "E-VALUES" in columns 1-8; and the letter "X" in column 23 if tabular printout is not desired. For SYMAP Version 6, to have the reference names which can be punched on the E-VALUES card printed next to the listing of data point map locations and values, punch an "X" in column 72 of the E-VALUES package header card.

On the last card, punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, punch the values to be applied at the data points given in the B-DATA POINTS package or to the zonal outlines specified in the A-CONFORMOLINES package. Each value is punched on a separate card and a card must be provided corresponding to every data point or to every data zone. Be sure that cards are kept in proper sequence. Punch each value as a decimal number right justified in columns 11-20. If a value is negative, punch a minus sign before it.

If the same E-VALUES package is to be used for both a contour and a conformant map, the reference order established for points, zones and their associated values must be entirely consistent, and only one data point from each zone may be included in the B-DATA POINTS package. Otherwise (when and if both types of maps are desired), separate E-VALUES packages must be prepared; one to be associated with the A-CONFORMOLINES package (for the conformant map), the other to be associated with the B-DATA POINTS package (for the contour map).

2.8 E1-VALUES INDEX

The E1-VALUES INDEX package is used to change the reference order of data values in the E-VALUES package. Normally, the conformant outlines as specified in the A-CONFORMOLINES package and the data points as specified in the B-DATA POINTS package are assigned data values from the E-VALUES package in the order in which they are listed. The E1-VALUES INDEX package is used in rare cases and only to avoid some considerable difficulty in reordering a non-standard B-DATA POINTS, A-CONFORMOLINES or E-VALUES package.

The A-CONFORMOLINES package provides for specification, by punching or by coded implication, of the reference order number of the data value with which each outline is to be associated. This data value reference number may be altered by using the E1-VALUES INDEX package. The B-DATA POINTS package contains no provision for indicating reference order other than the sequence in which the locations of data points are specified. If the data values are not listed in precisely the same order as the data points with which they are to be associated, the reference order of the data values must be altered. This may be done by using the E1-VALUES INDEX package.

On the first card of this package, punch E1-VALUES INDEX" in columns 1-15.

On the last card, punch the number "99999" in columns 1-5.

On the other cards, to be inserted between the first and last, punch the current reference order number of the data value as an integer right justified in columns 1-5; then punch the replacement reference order number for that value as an integer right justified in columns 6-10. If columns 6-10 are left blank, the number following that assigned to the previous card is assumed.

To have the reference order number one (referring to the first value) replaced by the reference order number two (referring to the second value), only one card is needed (in addition to the first and last). Punch the number "1" in column 5 and the number "2" in column 10. In this way the first card in the E-VALUES package will now be associated with the second data point in the B-DATA points package.

To have reference to the fifth value replaced by reference to the fourth value, provide four cards with the reference numbers "1" through "4" successively punched in column 5, followed by one card with the number "5" punched in column 5 and the number "4" punched in column 10. If another card with the number "6" punched in column 5 (and columns 6-10 left blank) were to follow, reference to the sixth value would be replaced by reference to the fifth value, since the preceding card had specified reference to the fourth value.

If a card is included on which columns 1-5 are left blank, this referenced value--the reference number of the previous value plus one--will be treated as missing data for the map.

2.9 F-MAP

This package instructs the computer to make a map--based on the information supplied in the prior packages--and is used to specify the precise form of that map in terms of certain available optional treatments known as electives.

On the first card of this package, punch "F-MAP" in columns 1-5; and the letter "X" in column 23 if printout of the input data is not desired.

On the last card, punch the number "99999" in columns 1-5.

On the second, third, and fourth cards, punch the title you wish to have appear below the map in columns 1-72. Be sure your title is clearly descriptive to differentiate the particular map being requested from all other maps, of a similar nature, which have been run previously or which may be run in the future. One or more of these three cards may be left blank if desired, but all three cards must be provided.

On other cards, to be inserted between the fourth and last cards, punch any "electives" that may be desired, as described in Section 3. Whenever a map elective is not specifically called for, the standard result described under each elective will automatically occur.

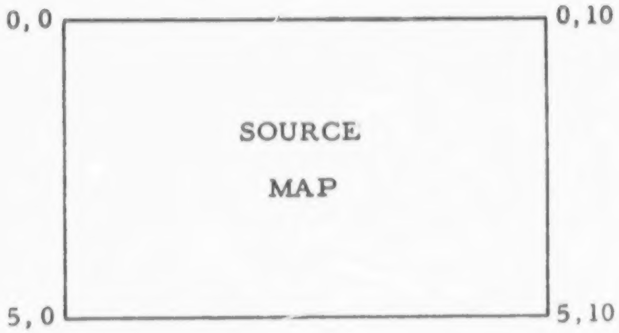
3. MAP PACKAGE ELECTIVES

The content, scale, tone, and numerous other features of the output map are specified by the user through the electives of the F-MAP package. All electives have a standard default and consequently a map can be produced without specifying any of the electives. Electives 1 and 2 control the size and scale and are used with almost all maps; electives 3 through 7 determine the value range, contour intervals, and symbolism, and are frequently used. Elective 8 suppresses the white contour lines and elective 10 provides for descriptive text below the map. All other electives are used less frequently and are more specialized.

3.1 ELECTIVE NUMBER 1
SIZE OF OUTPUT MAP (1 card)

To specify the size of the output map, punch the identifying elective number "1" in column 5. Punch the vertical dimension of the rectangular map border (the distance from the middle of the top border to the middle of the bottom border) in inches as a decimal number right justified in columns 11-20. Then punch the horizontal dimension of the rectangular map border (the distance from the middle of the left border to the middle of the right border) in inches as a decimal number right justified in columns 21-30, as shown in Example 3-1. Dimensions should be carried to at least one decimal place either whole or fractional. Whatever vertical dimension is specified will be rounded to the nearest one-eighth inch; whatever horizontal dimension is specified will be rounded to the nearest one-tenth inch.

EXAMPLE 3-1



Elective 1--size of map--for source map shown and output map of same size, would be:

Column:	1-5	6-10	11-20	21-30
	1		5.	10.

NOTES:

This elective specifies in inches the dimensions of the map which will be produced--map content plus rectangular border. The vertical dimension includes 1 character location (.125 inches) for the top & bottom borders combined. Therefore, the map content will be .125 inches less than the dimension specified. Likewise, the horizontal dimension includes 1 character location(.1) for both left and right borders combined and the SYMAP content will be .1 inches less than the dimension specified. For example, the horizontal dimension of 10.0 inches specified will result in a map content of 9.9 inches, or 99 columns, and a map border of 0.1 inches (1 column); the vertical dimension of 5.0 inches will result in a map content of 4.875 inches, or 39 rows, and a map border of 0.125 inches (1 row).

It is important to note that the dimensions of the border specified must conform to the dimensions of the rectangular border originally drawn on the source map and used to measure the x and y coordinates for the various packages; otherwise a scale change will occur between the source map and the output map. (The SYMAP program calculates the map scale by dividing the number of inches specified in Elective 1 by the number of source map units specified in Elective 2 -- The difference between extreme minimum and extreme maximum points; if either or both electives are omitted, the calculation is based on the standard for each elective).

Generally, the portion of the source map that is to form the content of the output map must be determined and specified in Elective 2 before the size of the output map can be specified in Elective 1. The measurements of the minimum and maximum extreme points that define this content--(0,0) and (5,10) in this case--should be in the same units as used for all coordinate locations, unless row and column measurements are being used: then, Elective 2 must be specified in inches. Elective 1 is always specified in inches of output map.

To specify changes in the border size without affecting the scale of the map, use Elective 14 - Margin; to specify a change in map scale without re-calculating Elective 1, use Elective 13 - Scale.

Example 3-1
Cont'd

If a horizontal dimension greater than 13 inches is specified, the map will be printed in two or more sections or panels, for mounting side by side. There is no set limit as to the border dimensions which may be specified. However, to safeguard the user who makes a gross dimensional error, the program prevents any dimension of the map from exceeding 72. 0 inches, unless elective 16 is specified to remove this safeguard. If elective number 16 is not specified, and if one or both dimensions exceeds 72. 0 inches, the larger dimension will be reduced to 13. 0 inches and the other dimension will be reduced proportionately.

It is possible to leave out either the vertical or the horizontal dimension. Whichever dimension is omitted will be calculated by means of the other electives--either specified or standard.

If elective number 1 is not included, the program will calculate the relative vertical and horizontal dimensions on the basis of extreme point locations and assign 13. 0 inches to the larger dimension.

Standard is 13. 0 inches for the larger dimension of the map, with the smaller dimension proportioned accordingly.

3. 2 ELECTIVE NUMBER 2
EXTREME POINTS OF INPUT MAP (1 card)

This elective is used to specify the coordinate location of the two extreme points on the source map which determine the content of the map to be produced.

The content may include all or any portion of the area shown on the source map.

If the two extreme points are not specified, the program will establish them by selecting the minimum and maximum coordinates from a preceding package--from the A-CONFORMOLINES package for a conformant map; from the A-OUTLINE package or, if omitted, from the B-DATA POINTS package for a contour or proximal map; or, if all these are omitted, from the C-OTOLEGENDS package.

To establish the two extreme points to be specified, first designate the area to be mapped by a rectangle drawn on the source map. The coordinate locations of the upper left corner and the lower right corner of this rectangle are, respectively, the minimum and maximum extreme points which will determine the content of the map to be produced. Measure these coordinate locations from the top and left borders originally established on the source map and specify them in the scale used for all coordinate locations. Note that if row and column measurements have been used, the coordinate locations for this elective must be specified in inches. Negative coordinates may be specified in order to include any portion outside the study area. Enter each coordinate as a decimal number right justified in the columns assigned.

On one card punch the identifying elective number "2" in column 5; the minimum vertical coordinate in columns 11-20; the minimum horizontal coordinate in columns 21-30; the maximum vertical coordinate in columns 31-40; the maximum horizontal coordinate in columns 41-50, as shown in Example 3-2.

Standard is to have the program select the extreme points which determine the content of the map. Use of this elective is recommended together with elective number 1 (size), and also with elective number 13 (scale) and elective number 14 (shift) if either or both of these electives are used.

3.3 ELECTIVE NUMBER 3
NUMBER OF LEVELS (1 card)

To specify the number of levels or class intervals into which the total value range is to be divided for designation by symbolism, punch the identifying elective number "3" in column 5, followed by the number of levels desired--from 2 through 10--punched as a decimal number right justified in columns 11-20, as shown in Example 3-3.

Standard is five levels.

3.4 ELECTIVE NUMBER 4
VALUE RANGE MINIMUM (1 card)

To specify a value to be used as the minimum value of the total value range, punch the identifying elective number "4" in column 5, followed by the minimum value desired punched as a decimal number right justified in columns 11-20. If values below this minimum are provided in the E-VALUES package, an area of symbolism

EXAMPLE 3-2

Elective 2--extreme points of input map--for the same source map shown in Example 3-1 would be:

Column	1-5	6-10	11-20	21-30	31-40	41-50
	2		0.	0.	5.	10.

EXAMPLE 3-3

Using 3 levels with values ranging from a minimum of 0 to a maximum of 30, the first level is to contain values from 0 to 10, the second level, values from 10 to 20, and the third level, values from 20 to 30.

Electives 3 through 6 for this example would be:

Column	1-5	6-10	11-20
	3		3.
	4		0.
	5		30.

Elective 6 would not be needed since the standard is to have each interval assigned an equal range.

consisting of the letter "L" (for LOW) will appear on the map.

Standard is to use the minimum value of the data.

3.5 ELECTIVE NUMBER 5
VALUE RANGE MAXIMUM (1 card)

To specify a value to be used as the maximum value of the total value range, punch the identifying elective number "5" in column 5, followed by the maximum value desired punched as a decimal number right justified in columns 11-20. If values above this maximum are provided in the E-VALUES package, an area of symbolism consisting of the letter "H" (for HIGH) will appear on the map.

Standard is to use the maximum value of the data.

3.6 ELECTIVE NUMBER 6
VALUE RANGE INTERVALS (1 or 2 cards)

This elective permits you to specify the range for each level or interval into which the total value range of the data (or the total range as modified under electives 4 and 5) is to be divided.

Standard is to have each level or interval assigned an equal range.

To specify the range for each level so that the data points will be distributed equally among levels (i.e., in so far as possible, to have the same number of data points appear in each level), punch the identifying elective number "6" in column 5. It is not advisable to use this option if greater than 20% of the data points all have the same value.

To specify any desired range for each level, punch the identifying elective number "6" in column 5, followed by numbers proportionate to the size of the desired ranges. These should be punched as decimal numbers right justified in columns 11-20 for the first level, 21-30 for the second level, 31-40 for the third level, etc., as shown in Examples 3-4 and 3-5. Only if there are more than 6 levels, continue on a second card, punching the number for the seventh level in columns 11-20, for the eighth level in columns 21-30, etc. --for a maximum of ten levels. Column 5 is left blank on the second card, as shown in Example 3-6.

3.7 ELECTIVE NUMBER 7
SYMBOLISM (5 cards)

This elective permits one to specify symbolism other than standard. If this elective is used, all 5 cards must be included.

On the first card punch the identifying elective number "7" in column 5.

On the second card punch in the appropriate columns the basic characters desired.

EXAMPLE 3-4

If the data, as provided in the E-VALUES package, is to be divided into five levels (which is standard) from a low value of 100 to a high value of 500 as follows:

level	value range
1	100-200
2	200-300
3	300-350
4	350-400
5	400-500

either the actual value range for each level or the proportion or ratio of the total range which should be assigned to each level may be specified in elective number 6:

Column	1-5	11-20	21-30	31-40	41-50	51-60
actual	6	100.	100.	50.	50.	100.
porportion (%)	6	25. 0	25. 0	12. 5	12. 5	25. 0
ratio (x/8)	6	2.	2.	1.	1.	2.

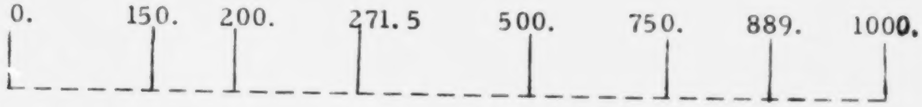
EXAMPLE 3-5

If the data, as provided in the E-VALUES package, is to be divided into four groups--the lowest 10%, the next 25%, the next 35%, and the remainder--the one card for elective number 6 would be punched as follows:

Column	1-5	11-20	21-30	31-40	41-50
	6	10.	25.	35.	30.

EXAMPLE 3-6

To specify these value range intervals--



--punch the two cards for elective number 6 as follows:

Column	1-5	11-20	21-30	31-40	41-50	51-60	61-70
Card 1	6	150.	50.	71.5	228.5	250.	139.
Card 2		111.					

A value on the border between two levels will always be assigned to the higher level. For instance, the value 150.0 would be in the second level; the highest value in the first level would be 149.99, assuming a rounding to two decimal places. The exception to this rule occurs in the highest level where the maximum value--here, 1000.0--would be assigned to that level since no higher level exists. Electives 3, 4, and 5 would also be needed for this example:

Column	1-5	11-20
	3	7.
	4	0.
	5	1000.

On the third, fourth and fifth cards punch in the appropriate columns any overprint characters desired. (If no overprinting is desired, these three cards will be blank.)

In punching the second, third, fourth, and fifth cards, columns 1-10 are used to specify the general symbolism for each level (column 1 for the symbol to designate the first level, column 2 for the symbol to designate the second level, etc. --for as many levels as are to be used). Columns 11-20 are used to specify the respective data point symbols (column 11 for the symbol to designate data points in the first level, column 12 for the symbol to designate data points in the second level, etc.).

Standard general symbolism varies according to the number of levels specified in elective number 3. Table 3-1 shows the number of levels to be mapped, the general symbolism which will appear at each of those levels, and the corresponding data point symbols.

Additional distinctive symbolism may be specified in columns 21-29. The standard symbolism which will occur includes:

Column 21	low, general symbolism-- "L", and three blanks
22	low, data point symbol-- "L", ".", and two blanks
23	high, general symbolism-- "H", and three blanks
24	high, data point symbol-- "H", "H", "H", and "/"
25	background symbolism (beyond the study area outline)--all blanks (or white)

TABLE 3-1

STANDARD SYMBOLISM FOR VARIOUS LEVELS

	general symbolism										data point symbolism									
column:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
number of levels desired:																				
1	.										1									
2	.	■									1	2								
3	.	O	■								1	2	3							
4	.	+	O	■							1	2	3	4						
5	.	+	O	⊖	■						1	2	3	4	5					
6	.	+	X	⊖	⊠	■					1	2	3	4	5	6				
7	.	'	+	X	⊖	⊠	■				1	2	3	4	5	6	7			
8	.	'	+	X	O	⊖	⊠	■			1	2	3	4	5	6	7	8		
9	.	'	=	+	X	O	⊖	⊠	■		1	2	3	4	5	6	7	8	9	
10	.	'	-	=	+	X	O	⊖	⊠	■	1	2	3	4	5	6	7	8	9	*

26	contour line or conformant boundary symbolism--all blanks (or white)
27	no data (used only for problems arising from the specification of barriers)--"N", and three blanks
28	superimposed data points--"S", and three blanks
29	data points with invalid values, so-called "missing data"--"M", and three blanks

Table 3-2 shows, as an example, the characters that would be punched on the second, third, fourth, and fifth cards as standard for ten levels and for distinctive symbolism. Examples 3-7, 3-8, and 3-9 depict three illustrations of non-standard symbolism for widely different needs.

3.8 ELECTIVE NUMBER 8
SUPPRESSION OF CONTOUR LINES (1 card)

To suppress white contour, conformant, or similar lines between adjacent levels of symbolism, punch the identifying elective number "8" in column 5.

Standard is to show such lines, insofar as space between data points may be adequate to show them without loss of descriptive symbolism.

3.9 ELECTIVE NUMBER 9
SUPPRESSION OF HISTOGRAM BARS (1 card)

To suppress the histogram bars which show graphically the frequency distribution of data point values in each level, punch the identifying elective number "9" in column 5. Standard is to have histogram bars.

TABLE 3-2

SYMBOLISM FOR LEVELS AND SPECIAL PURPOSES

Column	Description		Standard Symbolism			
		Card no:	2	3	4	5
1	General symbolism for level:	1	.			
2		2	'			
3		3	-			
4		4	=			
5		5	+			
6		6	X			
7		7	O			
8		8	O	-		
9		9	O	X		
10		10	O	X	A	V
11	Respective data point symbols for level:	1	1			
12		2	2			
13		3	3			
14		4	4			
15		5	5			
16		6	6			
17		7	7			
18		8	8			
19		9	9			
20		10	*			
21	Low--general symbolism		L			
22	Low--data point symbolism		L	.		
23	High--general symbolism		H			
24	High--data point symbolism		H	H	H	/
25	Background symbolism					
26	Symbolism for contour lines					
27	No data (used only with barriers)		N			
28	Superimposed data points		S			
29	Data points with invalid values		M			

EXAMPLE 3-7

The following illustration shows the correct punching of the 5 cards required for elective number 7, if 6 levels are specified in elective number 3, the reverse order of standard symbolism is desired, and you wish the general symbolism to appear at the data point locations.

```
                 11111111112222222
Column  12345678901234567890123456
----------------------------------
card 1      7
card 2  OOOX+     OOOX+.
card 3  XX-       XX-
card 4  A         A
card 5  V         V
```

EXAMPLE 3-8

The following illustration shows the correct punching of the 5 cards required for elective number 7, if you wish to: suppress all general symbolism, have the contour lines black, and have a background of dashes--while keeping everything else standard.

```
                 11111111112222222
Column  12345678901234567890123456
----------------------------------
card 1      7
card 2            12345         -O
card 3                           X
card 4                           A
card 5                           V
```

EXAMPLE 3-9

The following illustration shows the correct punching of the 5 cards required for elective number 7, if you are running a proximal type map and wish to use the symbols A, B, C, D and E for the data classes.

Column	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
card 1					7																					
card 2	A	B	C	D	E						A	B	C	D	E											
card 3											/	/	/	/	/											
card 4																										
card 5																										

3.10 ELECTIVE NUMBER 10
TEXT (3 to 32 cards)

This elective permits you to display below the map supplementary information, in addition to the legends and the map title.

On the first card punch the identifying elective number "10" in columns 4-5.

On the last card punch the number "9999" in columns 1-4. These four nines show the end of the text and should not be confused with the five nines required to end packages.

On not more than 30 other cards--to be inserted between the first and last--punch in columns 1-72 any supplementary information likely to be helpful for future reference, such as the source and date of data, or the name of the person running the map.

Standard is to have no text, but this will seldom be desirable.

3.11 ELECTIVE NUMBER 11
PRINTING ACTUAL VALUE AT EACH DATA POINT (1 card)

To have each data value appear at its data point location, punch the identifying elective number "11" in columns 4 and 5. The numeric values will appear with two decimal places, the decimal points being located at the data points specified in the B-DATA POINTS package. Up to ten digits may appear to the left of the decimal point. Standard is to have the data point symbol appear.

3.12 ELECTIVE NUMBER 12
MULTIPLE ELECTIVE REPEAT (1 card)

This elective is available for the convenience of the user in preparing the input for more than one map to be produced by the computer in a single run, as discussed in Section 4.

Elective number 12 may be used to specify that in a second or subsequent map of a single submission, all non-standard electives used in the preceding map are to be repeated. Non-standard electives may be added, or modified and reinserted, in sequence, but this elective may not be used if any non-standard electives are to be deleted, as shown in Example 3-10.

To repeat all non-standard electives used in the preceding map, punch the identifying elective number "12" in columns 4-5.

Standard is to provide the required cards for all non-standard electives to be used in each map.

3.13 ELECTIVE NUMBER 13
SCALE (1 card)

To specify the scale of the output map, punch the identifying elective number "13" in columns 4-5, followed by the scale, i.e., the number of inches, or

EXAMPLE 3-10

The following illustration shows the correct punching of the cards needed to run two maps from the same set of data. In the first map, an equal number of data points in each level is specified using elective 6. In the second map, the number of levels is changed, an equal range is assigned to each level (the standard for elective 6), and the contour lines are suppressed using elective 8.

```
F-MAP
TEST MAPS
EXAMPLE 3-10
MAP ONE
    1           10.       10.
    2            0.        0.        10.        10.
    3            6.
    4            0.
    5          100.
    6
99999
F-MAP
TEST MAPS
EXAMPLE 3-10
MAP TWO
    3            4.
    6           25.       25.        25.        25.
    8
   12
99999
```

Note that by using elective 12 to repeat non-standard electives, reversion to standard can only be accomplished by respecification, as in elective 6, and not by omission. Omission results in the use of the same non-standard specification as the first map--electives 1, 2, 4, and 5. Electives may be modified--elective 3--and added--elective 8.

fraction thereof, desired to represent one source-map measurement unit, punched as a decimal number right justified in columns 11-20.

If the scale of the map is specified as "1.0", each inch on the output map will represent one source-map measurement unit. If the source-map measurement unit is in inches, these two maps will be at the same scale. To increase the scale of the output map, increase the specified number; to decrease the scale of the output map, decrease the specified number--0.5 in columns 11-20 will produce an output map one half the size of the source map, assuming the source map was measured in inches. Often this elective can be used in conjunction with elective 2 (extreme points) and in place of elective 1.

If not specified, the scale will be calculated from the size of map and from extreme points--as specified in electives 1 and 2, or from the standards of these electives. If this cannot be done, a scale of "1.0" will be assigned.

Standard is to have the scale established automatically from other electives.

3.14 ELECTIVE NUMBER 14
MARGIN (1 card)

This elective is used to obtain a margin between the extreme points of the study area and the corresponding output map borders. It may also be used to narrow the content of the output map.

The program normally positions the content of the map--as determined by the extreme points of the study area--so as to fit precisely within the map border. Specifying a positive shift causes the program to assume a border closer to the center of the map by the amount specified. A margin results from the discrepancy between the real border used and this assumed border. Inversely, a negative specification assumes that the border should be further from the center than the real border, and the content of the map will be truncated along the real border by the amount specified.

The scale of the study area is subject to change depending on the border shifts specified. If the scale is specified in elective number 13, this will override any change in scale which might have resulted from a border shift.

On one card, punch the identifying elective number "14" in columns 4-5; punch the desired shift for each border, specified in inches, as a decimal number right justified in the columns assigned as follows:

Top border, columns 11-20
Left border, columns 21-30
Bottom border, columns 31-40
Right border, columns 41-50.

To obtain a half-inch margin between the study area (as determined by the extreme points) and all four map borders, specify "0.5" in each column field. Symbolism will be printed in this margin according to the specification in column 26 of elective 7, with white, or blank, being the standard. To have no shift occur, specify "0.0". To narrow the content of the map, by truncating the study area one half-inch along any border, specify a shift of "-0.5" in the appropriate column field.

Standard is to have extreme points of the study area "touch" their corresponding map borders.

3.15 ELECTIVE NUMBER 15
NUMBER OF OUTPUT CHARACTERS PER INCH (1 card)

To specify the number of characters per inch--vertically and horizontally--at which the map will be printed, punch the identifying elective number "15" in columns 4-5; punch the number of lines per inch (vertically) as a decimal number right justified in columns 11-20; and the number of columns per inch (horizontally) as a decimal number right justified in columns 21-30.

This elective is primarily for persons who may need to use a printer other than one which prints at 8.0 lines per inch. Output prepared for printing at 8.0 lines per inch will appear somewhat overextended when printed at 6.0 lines per inch. By specifying 6.0 lines per inch, fewer lines of output will result, but the overall size of the map will be the same. The column field provided for changing the number of columns per inch need not be, and rarely would be, used, since almost all computer printers print at 10.0 columns per inch.

Normally, printing is at 6.0 lines per inch on standard chain-line printers, but paper "carriage tapes" can generally be prepared to permit printing at 8. 0 lines per inch and avoid the necessity of using this elective.

Standard is eight lines per inch and ten columns per inch.

3. 16 ELECTIVE NUMBER 16
LARGE SIZE MAPS (1 card)

To obtain a map whose vertical or horizontal dimension exceeds 72. 0 inches, punch the identifying elective number "16" in columns 4-5.

This elective is used to nullify a built-in safeguard which prevents the printing of a map larger than 72. 0 inches in either dimension. The safeguard provides a measure of protection for the user who may have made a potentially very costly error in specifying the size of the map. Elective number 16 should be used with caution, and with exceeding caution if the user is making a lengthy run.

If this elective is not specified, and if one or both dimensions exceeds 72. 0 inches, the larger dimension will be reduced to 13. 0 inches and the other dimension (and the scale of the map) will be reduced proportionately.

Standard is a map not exceeding 72. 0 inches in either dimension; or, a map reduced to 13. 0 inches if a dimension exceeding 72. 0 inches was erroneously specified in elective number 1.

3.17 ELECTIVE NUMBER 17
SUPPRESSION OF TABULAR OUTPUT DATA FOR CONFORMOLINES OR DATA POINTS (1 card)

To suppress the tabular printout of output data including the location, value, and level assigned to each conformoline or data point, punch the identifying elective number "17" in columns 4-5.

This printout should not be confused with the tabular printout of the input data, for each package, which may be suppressed by punching an "X" in column 23 on the first card of A-CONFORMOLINES and B-DATA POINTS packages.

Standard is to have a tabular printout of the output data immediately preceding the map.

3.18 ELECTIVE NUMBER 18
INVALIDATION OF MINUS ZERO AND BLANK VALUES (1 card)

To have minus zero (-0. 0) and blank data values considered invalid for computation, punch the identifying elective number "18" in columns 4-5.

At any data point whose associated value is considered invalid, the alphabetic symbol "M" will appear in place of the numerical symbol which would normally indicate the value level assigned to that point. The appearance of invalid data point symbols may be eliminated through the use of elective number 23.

Over any conformoline zone whose associated data value is considered invalid, the standard (blank) background symbolism will replace the distinctive general symbolism which would normally indicate the value level assigned to that zone.

Invalid data symbolism other than standard may be specified in elective number 7. The IBM 360 does not distinguish between 0. 0 and -0. 0 or blanks; consequently this elective is inoperable on the IBM 360, and invalid data cannot be specified by this method.

Standard is to have all data values considered valid, and used for calculation, including unspecified (blank) values which will be assigned the value of "-0. 0".

3.19 INVALID DATA RANGE ELECTIVES

3.19.1 Elective Number 19
Minimum Value for Invalid-Data Range (1 card)

To specify a minimum value for the range of invalid data so that data values which are equal to or more than the specified minimum will be considered invalid and not used for calculation, punch the identifying elective number "19" in columns 4-5, and punch the desired minimum as a decimal number right justified in columns 11-20.

The standard symbolism for invalid data and possibilities for the modification of such symbolism are described under elective number 18.

Standard is to have all data values considered valid and used for calculation.

3.19.2 Elective Number 20
Maximum Value for Invalid Data Range (1 card)

To specify a maximum value for the range of invalid data so that data values which are equal to or less than the specified maximum will be considered invalid and not used for calculation, punch the identifying elective number "20" in columns 4-5 and punch the desired maximum as a decimal number right justified in columns 11-20. The standard symbolism for invalid data and possibilities for the modification of such symbolism are described under elective 18.

Standard is to have all data values considered valid and used for calculation.

3.19.3 Electives 19 and 20 in Combination (2 cards)

If the same value is specified for the minimum invalid value and for the maximum invalid value, only that value will be considered invalid and will not be used for calculation when encountered.

If the specified minimum invalid value is less than the specified maximum invalid value, data values between and equal to the two values will not be used for calculation. If the specified minimum invalid value is greater than the specified maximum invalid value, only data values between the two values will be used for calculation.

Standard is to have all data values considered valid and used for calculation.

3.20 ELECTIVE NUMBER 21
STORE LOCATION VALUES FOR MAP ON TAPE
(1 card)

As a map is produced, the program calculates a value for every print location used in displaying the final map. This elective stores these values on tape for later input to other programs. The resulting tape cannot be used for printing additional copies of the normal SYMAP.

When this elective is specified, the following steps occur:

i) Tape unit 8 is rewound if it is the first map.

ii) At the beginning of the map, a 26-word binary record is written (on tape 8) without format control. The words contain the following information: (a) the fixed-point number of rows for the entire map (which is exactly equal to the number of rows between, and exclusive of, the top and bottom borders, plus one); (b) the fixed-point number of columns for the entire map (which is exactly equal to the number of columns between, and exclusive of, the left and right borders, plus one); (c) the floating-point coordinate of the top border; (d) the floating-point coordinate of the bottom border; (e) the floating-point number of rows per coordinate unit; (f) the floating-point number of columns per coordinate; (g) the floating-point value of the minimum of the level range; (h) the floating-point value of the maximum of the level range, and 18 additional words, whose content is irrelevant to the user.

iii) During the creation of a rectangular shaped contour or proximal map with no OUTLINE package (and which is not a "base map" where all values are identical), a series of additional binary records are written. Each word is a floating-point value and corresponds to the value calculated for a map location. Each record corresponds to one row of one strip (printout paper strip) of the map. The records are produced in the same order as the map is printed. Each word of the record corresponds to the respective print location within the map, excluding border print locations.

iv) At the end of the map, an end-of-file is written on tape 8. For example, if a contour map is run at the size of 10. 0 x 10. 0 inches, the number of rows will be 80. (Numbering of map rows begins with the top border as zero, and similarly with the left border for columns. Measurements are always taken from the center of the top row (border)--similarly for columns. As a result, each border may be thought of as 1/2 unit in print character height, or width. Thus, the 80 rows are comprised of 79 rows of values plus the top and bottom border of the map which are rows 0 and 80, respectively.) Likewise, the number of columns will be 100. There will be 79 records of 99 values each written on the tape in addition to the first 26-word record. These will be followed by an end-of-file.

To store the map location values on a tape for later use with other programs--particularly for input to the SYMVU three-dimensional mapping program--punch the identifying elective number "21" in columns 4-5.

The following suboptions are available in SYMAP Version 6 (only the first one is in Version 5):

Column 11-20 blank	the standard suboption is the writing of the interpolated values onto a save tape for SYMVU
Column 11-20 > 0.	interpolated values will be printed as well as written on tape
Column 21-30 > 0.	row and column data point coordinates will be punched in format for SYMVU
Column 31-40 > 0.	values on tape are the maximum in each level instead of actual values
Column 41-50 > 0.	point legends will be punched on cards in format for SYMVU
Column 51-60 > 0.	contour points will be punched in format for SYMVU. Format is (3I10): row, column, symbol = 11.

Standard is not to print the values or write them on tape.

3.21 ELECTIVE NUMBER 22
CONTINUOUS CONTOURS (1 card)

This elective is provided for the user whose primary interest is contour lines. When the space between data points or zones is inadequate for the display of both symbolism and contour lines, the contour lines would otherwise be suppressed to permit the display of descriptive symbolism. Punch the identifying number "22" in columns 4-5 to ensure the appearance of contour lines.

Standard is to suppress contour lines when a loss of descriptive symbolism may result from their inclusion.

3.22 ELECTIVE NUMBER 23
SUPPRESSION OF INVALID DATA POINT SYMBOL (1 card)

To suppress the symbol--standard or as specified in elective 7--which appears at any data point whose associated value is considered invalid, resulting from the use of elective 18, 19, or 20, punch the identifying elective number "23" in columns 4-5.

Standard is to have the symbol "M" appear at any data point having an invalid data value.

3.23 ELECTIVE NUMBER 24
SUPPRESSION OF NUMERIC INTERPRETATION (1 card)

To suppress the printing of the numeric interpretation following the map, which includes:

> "ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL"
>
> "PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL",

punch the identifying elective number "24" in columns 4-5.

Standard is to have this numeric interpretation appear after the map, immediately preceding the frequency distribution chart.

3.24 ELECTIVE NUMBER 25
SUPPRESSION OF DATA POINT SYMBOLS (1 card)

To have data points, as specified in the B-DATA POINTS package, not appear within the zonal outlines on a conformant map, punch the identifying elective number "25" in columns 4-5. When this elective is not used, each specified data point will be designated by a symbol corresponding to the value level or class assigned to the zone in which it is located.

Standard is to have data points appear on a conformant map.

3.25 ELECTIVE NUMBER 26
OVERPRINT ALIGNMENT (1 card)

To correct the alignment of the overprint lines to coincide with the line to be overprinted, punch the identifying elective number "26" in columns 4-5.

With certain computers, the overprint line will appear one line below the original print line. This elective should be used only if this incorrect alignment is known to occur.

Standard is the automatic coincidence of print and overprint for the IBM 7094 Fortran Language and the IBM 360 for SYMAP Versions 5 and 6.

3.26 ELECTIVE NUMBER 27
DISTINGUISHING MAP TYPE (1 card)

To enable the program to identify the map desired as a contour map when data for both contour and conformant maps are included in the same submission, punch the identifying elective number "27" in columns 4-5.

Standard is to produce a conformant map if an A-CONFORMOLINES package has been included in the submission.

3.27 ELECTIVE NUMBER 28
DOT CONFORMANT MAP (1 card)
(available in SYMAP Version 6 only)

To produce a dot map when running a conformant map, punch the identifying elective number "28" in columns 4-5. The user may specify one of the following suboptions:

Column 11-20 blank	the standard suboption will dot a percentage of the area in a zone to a density corresponding to the level of that zone.
Column 11-20 > 0.	the number of dots in a zone will represent the value for that zone as a percentage of the total value range. No map levels will be produced.
Column 21-30 > 0.	each dot will have an absolute value equal to the number punched in these columns.

Standard is not to produce a dot map.

3.28 EXTRAPOLATION RANGE ELECTIVES

The program has the capability of extrapolating above the highest given value for a "hill"--or below the lowest given value for a "valley"--if the surrounding data indicates such a trend is reasonable. The following three electives control this range of extrapolation.

3.28.1 Elective Number 31
Relative Fractional Extrapolation (1 card)

To specify the fractional extrapolation, punch the identifying elective number "31" in columns 4-5, followed by the fractional extrapolation as a positive decimal number less than 1 punched right justified in columns 11-20. The computer will extrapolate above or below a local extreme at most this fraction of the total range of data values.

Standard is a fractional extrapolation of "0.1" of the total range of data values.

3.28.2 Elective Number 32
Absolute Extrapolation Minimum (1 card)

To specify the absolute extrapolation minimum, punch the identifying elective number"32" in columns 4-5, followed by the extrapolation minimum punched as a decimal number right justified in columns 11-20. The computer will extrapolate down and only as far as this value. Only where all the surrounding data points suggest a trend to this minimum, may the minimum value actually be reached.

This elective is recommended if the user knows a logical minimum for his data such as "0.0" for population density. If the fractional extrapolation and the data range are such that this minimum could not be exceeded anyway, this elective need not be used.

Standard is an extrapolation minimum equal to the minimum value of the data.

3.28.3 Elective Number 33
Absolute Extrapolation Maximum (1 card)

To specify the absolute extrapolation maximum, punch the identifying elective number "33" in columns 4-5, followed by the extrapolation maximum punched as a decimal number right justified in columns 11-20. The computer will extrapolate upward only as far as this value. Only where all the surrounding data suggests a trend to this maximum, may the maximum value actually be reached.

This elective is recommended if the user knows that his data values have a logical maximum such as "100.0" for most percentage data. If the fractional extrapolation and data range are such that this maximum could not be exceeded anyway, this elective need not be used.

Standard is an extrapolation maximum equal to the maximum value of the data.

3.29 SEARCH RADIUS ELECTIVES

The following two specialized electives allow the advanced user precise control over the selection of data points for interpolation. The search radius is the maximum distance the computer will search for data points to use in the interpolation process, in source map units.

3.29.1 Elective Number 34
Initial Search Radius (1 card)

To specify the initial search radius, punch the identifying elective number "34" in columns 4-5, followed by the initial search radius punched as a positive decimal number in columns 11-20. This elective will normally be used to preserve the initial search radius from a previous map after altering the number of data points or the locations of data points near the edge of the study area.

The standard bases the initial search radius on the number and spread of the data points, so that, on the average, seven data points will be found on the initial search.

3.29.2 Elective Number 35
Maximum Search Radius (1 card)

To specify the maximum search radius, punch the identifying elective number "35" in columns 4-5, followed by the maximum search radius punched as a decimal number in columns 11-20. The computer will, if necessary, search for data points up to this maximum search radius. The maximum search radius may not be less than the initial search radius.

The standard has no absolute search radius limit, but the computer will not search beyond the effective distance to the furthest data point.

3.30 ELECTIVE NUMBER 36
NUMBER OF DATA POINTS FOR INTERPOLATION (1 card)

To specify the minimum and the maximum number of data points on which interpolation is to be based, punch the identifying elective number "36" in columns 4-5; punch the desired minimum (0-10) as a decimal number right justified in columns 11-20, and the desired maximum (0-10) as a decimal number right justified in columns 21-30.

Standard is to have a minimum of four and a maximum of ten data points used for interpolation with an average of seven.

3. 31 ELECTIVE NUMBER 37
INDEPENDENCE (1 card)

To have the value at each character location computed independently, punch the identifying elective number "37" in columns 4-5.

This elective prevents smoothing across sharp lines created by barriers or by special uses of electives 35 and 36. Except for special maps, expenditure of the additional computer time required for the independent calculations specified by this elective is unnecessary.

Standard is to compute the values independently for only about one-sixth of the characters, from which values at intermediate characters are derived.

3. 32 PROXIMAL MAP ELECTIVES (3 cards)

The combination of electives 31, 36, and 37 is used to specify the proximal type of map. Include one card for each of the three electives. Punch the identifying elective numbers "31", "36", and "37" in columns 4-5 on successive cards. No other specification is required on these cards.

The proximal type of map is based on the "nearest neighbor" concept: the value and symbolism for each character location is determined by the value and symbolism applicable to the nearest data point. "Point" information is used for both the proximal and the contour type map--however, the intervening levels and interpolated surface which distinguish a contour map do not appear on a proximal map. The proximal map is especially useful for the display of qualitative data, particularly when the precise definition of zonal boundaries is not of paramount importance. When precise zonal boundaries are important, the conformant type map is recommended.

Standard--with the use of data by points--is the contour type of map.

3.33 ELECTIVE NUMBER 38
DATA GRAPH (1 card)
(Available in SYMAP Version 6 only)

To produce a scaled frequency distribution which will appear below the map, punch the identifying elective number "38" in columns 4-5.

Standard is not to produce the data graph.

3.34 ELECTIVE NUMBER 39
LINEAR REGRESSION PLANE (1 card)
(Available in SYMAP Version 6 only)

To produce a linear regression plane of the surface computed, punch the identifying elective number "39" in columns 4-5. The user may specify the following suboption:

Column 11-20 blank	the standard suboption is a default size 1/2 the width of the map. The maximum regression plane size is 13".

Column 11-20 $> 0.$	the number punched in these columns is the size factor, given as a fraction of the map width.

Standard is not to produce a linear regression plane.

4. SYMAP USER SUBROUTINE

Subroutine FLEXIN is an optional FORTRAN IV subroutine supplied by the user and used (1) to specify input data formats which differ from the standard SYMAP formats, and (2) to manipulate the data after it has been read. The most common use of this subroutine is with the E-VALUES package; if the data values are already on cards, tape, or disk, they can be read in according to the present format. It is not necessary to repunch or reformat the data to conform to standard E-VALUES package format.

The subroutine may be used with all of the packages except the F-MAP package. The A-OUTLINE, B-DATA POINTS, D-BARRIERS, E-VALUES, and E1-VALUES INDEX packages may be read in using formats specified in subroutine FLEXIN. On the other hand the A-CONFORMOLINES and C-OTOLEGENDS packages are always read in as usual using the standard formats even if subroutine FLEXIN is employed for manipulating the information. Manipulations can be performed on the information contained in any of these packages, once it has been read in.

The uses of subroutine FLEXIN described in the following sections apply mainly to the packages employed to produce contour maps. However, the same procedures also apply to conformant and proximal maps: E-VALUES packages are used for all three types of maps and the coordinates of the A-CONFORMOLINES packages can be manipulated in the same way as the C-OTOLEGENDS packages.

4.1 MULTIPLE MAP SUBMISSION

The multiple submission of maps can be accomplished quite easily with the SYMAP program. In the simplest case, several maps of the same variable may be desired, requiring changes only in the electives of the map packages. An F-MAP package is required for each map desired. Example 1-9 lists the five F-MAP packages required to produce the maps in Examples 1-8, 1-10, 1-11, 1-12, and 1-13. All of these maps use the same outline, data point, legend, and values packages. This illustrates that data packages are retained for successive maps unless changed or deleted.*

If maps are being made for the same study area and data point locations but for different variables, then only the E-VALUES and F-MAP packages need to be varied from map to map. If a single map is required for each of two variables, then the following order for the packages would be used:

A - OUTLINE
B - DATA POINTS
C - OTOLEGENDS
E - VALUES 1
F - MAP 1
E - VALUES 2
F - MAP 2.

When one or more maps are to be produced, additional cards must preceed the data packages. Even if no data or program manipulations are desired, subroutine FLEXIN must be provided in a "dummy" form since it is an integral part of the SYMAP deck. The dummy FLEXIN consists of the following three cards: (1) SUBROUTINE FLEXIN (IFORM, T, FIRST) (2) RETURN (3) END. These cards tell the computer that there are no special manipulations to take place, and, therefore, that all packages are in the standard formats read by the SYMAP program. Job control cards both preceed and follow the dummy subroutine; the data packages follow the second set of job control cards. These cards vary with the computing system and facilities used.*

---

* A package can be deleted by specifying only the first and last cards; this is shown in Example 1-14 for the C-OTOLEGENDS package.

Example 4-1 shows the dummy subroutine FLEXIN and the seven packages required to produce the single maps for each of two variables. Note that all the packages are on cards and in the standard format.

If the data for the E - VALUES packages were on tape or disk and the electives did not change from one F - MAP package to the next, then very few cards would actually be required after the first F - MAP package -- a minimum of 3 for the E - VALUES and 6 for the F - MAP package. When maps for a study area are to be run continuously over a long period of time, it would be advisable to store the outline and legends on a tape or disk file, store the data banks containing the data points and values on another file, and submit card decks for each run composed of only the map package, user subroutine and job control language.

If both contour and conformant maps are to be produced in the same run, there are several special considerations. When an A-CONFORMOLINES package is introduced, the program assumes that a conformant map is desired for that and all succeeding maps in the same series, and it will ignore A-OUTLINE and B-DATA POINTS packages. If a B-DATA POINTS package is to be used for the printing of data point symbols, elective number "25" must be specified in the F-MAP package of each map. On the other hand, unless or until an

---

* See Section 5 for a discussion of job control cards.

```
                                          · job control cards
      SUBROUTINE FLEXIN
      RETURN                              · dummy subroutine
      END

 A-OUTLINE                                · job control cards

                                          · package on cards
 99999
 B-DATA POINTS

                                          · package on cards
 99999
 C-OTOLEGENDS

                                          · package on cards
 99999
 E-VALUES

                                          · package on cards
 99999
 F-MAP

                                          · package on cards
 99999
 E-VALUES

                                          · package on cards
 99999
 F-MAP

                                          · package on cards
 99999
```

Example 4-1
Subroutine FLEXIN--Dummy

A-CONFORMOLINES package is encountered in a submission, the program assumes that a contour map is desired and all packages are used. Following the introduction of an A-CONFORMOLINES package, elective number "27" must be specified in the F-MAP package of each contour or proximal map desired. As a general safeguard when unrelated maps are to be run in a single submission, a card with the word "CLEAR" punched in columns 1-5 should be inserted before the first data package of any map which is unrelated to the preceding map. This will prevent any carry-over of data from a previous map.

In preparing the F-MAP package for any map, it is important to remember that non-standard electives are not automatically carried over from the preceding map. The standard form of each elective is employed for each map unless another specification is made in the F-MAP package. To repeat one or more non-standard electives used in the preceding map, include a separate card for each elective to be repeated. Punch the identifying elective number as an integer right justified in columns 1-5, and the word "SAME" in columns 6-9. To repeat all non-standard electives used in the preceding map, punch the identifying elective number "12" in columns 4-5. Non-standard electives may be added, or modified and reinserted in sequence, but elective number 12 may not be used if any non-standard electives are to be deleted by reversion to the standard.

4.2. DATA BANKS AND FORMATS

The most common use of subroutine FLEXIN is to extract a particular variable from a data bank for mapping. In this way it is not necessary to prepare a separate E-VALUES package in the standard format for each map to be produced.

Example 4-2 shows the simplest use of the subroutine. In this case only one package -- the first E-VALUES package -- is read in by the subroutine; all other packages are punched in the standard format and read in as usual by the program. In addition, the subroutine is used here only to read in the variable according to a non-standard format; no manipulations are performed.

In example 4-2 the subroutine instructs the program to read the variable T from *cards* (the unit *5* specified in the read statement is the card reader) according to the format found in statement number 100. This format says that the value will be found in columns 6-10 on each card. Each time that the Subroutine is called -- once for each data point -- a new value for the variable T will be read and returned to the main program as the value to be mapped for that data point.

The "READ" statement is used to identify, by means of a variable name (or acronym), each category of data *to be read* from a given data bank. The "FORMAT" statement is used to specify the exact location in the data bank of each data category specified in the READ statement. A "T" statement is often used to specify to the computer the data values *to be mapped*. This statement may simply direct the computer to map the values as read or to perform an arithmetic operation (manipulation) involving one or more categories of data and to map the values resulting from this computation. In Example 4-2 the variable name used is "T"; no manipulations are required, and therefore no separate "T" statement is necessary. If the variable name had been "VALUE",

```
                                                        job control cards
      SUBROUTINE FLEXIN(IFORM,T,FIRST)
      READ(5,100) T                                     subroutine
  100 FORMAT(5X,5F.0)
      RETURN
      END
                                                        job control cards
A-OUTLINE
                                                        package on cards
99999
B-DATA POINTS
                                                        package on cards
99999
C-OTOLEGENDS
                                                        package on cards
99999
E-VALUES          X
    1   10
                                                        package on cards
                                                          read by FLEXIN
99999
F-MAP
                                                        package on cards
99999
E-VALUES
                                                        package on cards
99999
F-MAP
                                                        package on cards
99999
```

Example 4-2
Subroutine FLEXIN--Non-Standard Format

for instance, then a "T" statement would have followed the "FORMAT" statement:

T = VALUE.

The user should refer to a basic FORTRAN IV manual for the acceptable variable names, format procedures, and other specifications that may be important to the writing of a Subroutine FLEXIN. These include:

-- columns used: The punching of the fortran statement begins in column 7; columns 1-5 are reserved for the statement number and column 6 for a code indicating that the statement occupies more than one card;

-- variable names: the variables are generally "floating point numbers" -- containing an actual or implied decimal point. The variable names may be acronyms which consist of any combination of letters and numerals such that the initial character is a letter other than I, J, K, L, M, or N, the total number of characters is six or less, and the resulting combination does not have some regularly established meaning; acronyms to be avoided include END, FORMAT, GOTO, IF, READ, RETURN, REWIND, and WRITE. The FORMAT statement is used to specify where on each data card in the E-VALUES package the named variable will be found. In the statement the "5X" directs the computer to skip 5 columns and to read the value which follows. "F5. 0" informs the computer that each of the values specified in this location is a "floating point number", that this number occupies a "field" of five columns, and that no digits follow the decimal point.

-- arithmetic expressions: when the variable (or variables) require manipulation prior to mapping, the following characters should be used for arithmetic operations:

+	for addition
-	for subtraction
*	for multiplication
/	for division
**	for exponentiation.

Arithmetic operations will be carried out at three levels and in this sequence: first, exponentiation; second, multiplication and/or division; third, addition and/or subtraction. Within each level, operations are carried out from left to right for a given arithmetic statement. Use balanced parentheses to alter this sequence of operations between levels or to establish the sequence of operations between multiplication and division, or addition and subtraction, when necessary.

In Example 4-2 all but the first E-VALUES package are read in using the standard format. It is necessary to inform the program that this E-VALUES package is to be treated differently.

On the first card of this package "E-VALUES" is punched in columns 1-8 and the letter "X" in column 18[*]; this tells the computer that a continuation card follows with additional instructions.

On the second card -- the continuation card -- the subroutine FLEXIN identifying number (an integer from 1 to 25) is punched right justified in columns 1-5. This number is read by the subroutine and becomes the value for the variable "IFORM". If IFORM =0, the subroutine will not be called. In Example 4-2, this variable is used only to "call the subroutine" when the first E-VALUES package is read in by the program. The data bank on cards immediately follows this second card, one card -- or group of cards -- for each data point value. The total number of data points is punched as an integer number right justified in columns 6-10. In Example 4-2 IFORM = 1 and the number of data points is 10. This means that 10 values for the variable "T" will be read in one by one from the data bank following the second card.

---

[*] This can be any non-blank character in columns 16-20; by convention an "x" in column 18 is generally used.

On the last card the number "99999" is punched in columns 1-5 to signal the end of the package.

When Subroutine FLEXIN is called, three variables -- IFORM, T, and FIRST -- are transferred between the subroutine and the main program. The variables IFORM and FIRST are carried into the subroutine as control variables. The data value to be mapped is carried back to the main program as T. Subroutine FLEXIN is called once for every value to be mapped and a value for T is returned each time.

FIRST is a logical variable; it must be declared "type logical" by the statement, LOGICAL FIRST. It is "true" the first time subroutine FLEXIN is called for a given package and "false" on all other entries from the main program. This is set outside the subroutine. It is useful for initializing a counter and particularly for rewinding a tape or disk file so that the first entry will be read the first time the subroutine is called for a given package.*

---

*The use of the variable name, FIRST, is coincidental with the fact that the variable is set "true" the first time the subroutine is called.

In Example 4-2 the subroutine was used to read in or manipulate only one package -- the first E-VALUES package. In Example 4-3 the identical two maps are to be produced using the same outline, data point, legends, and map packages. However, the E-VALUES packages are different:

(i) the variables for both maps are contained in the same data bank.

(ii) the variable for the first map is derived in the subroutine by reading in two variables and dividing one by the other: T = A/B.

(iii) because the variables are in one data bank, provided on cards with the first E-VALUES package, the second E-VALUES package does not contain any data.

Subroutine FLEXIN is used to read and manipulate both of the E-VALUES packages. The variable IFORM is used to distinguish between packages which are to be read and manipulated if there is more than one. In Example 4-3 the value of IFORM on the second card of the first E-VALUES package is "1"; IFORM = 2 for the second E-VALUES package. Notice that the first E-VALUES package is the same as in Example 4-2: the number of data points to be read in, 10, is punched right justified in columns 6-10 and the 10 data cards follow this second card.

The second E-VALUES package is different: no cards follow the second card since the data values are to be read in from a tape or disk file as specified by the instructions in the subroutine. The "x" in column 13 of the second card causes the tape or disk file to be

```
                                            • job control cards
      SUBROUTINE FLEXIN(IFORM,T,FIRST)
      LOGICAL FIRST                         • subroutine
      GO TO(1,2),IFORM
    1 CONTINUE
      IF(FIRST) REWIND 12                   • used with first values
      READ(5,10) A,B,C                        package
   10 FORMAT(3F10.0)
      WRITE(12)C
      T=A/B
      RETURN
    2 CONTINUE
      IF(FIRST)REWIND 12                    • used with second values
      READ(12)C                               package
      T=C
      RETURN
      END

                                            • job control cards
                                            • outline, data point, and
E-VALUES          X                           legend  packages on cards.
    1   10

99999                                       • package on cards;
F-MAP                                         read by FLEXIN

                                            • package on cards
99999
E-VALUES          X
    2   10   X                              • package on tape or disk
99999                                         (unit 12) read by FLEXIN
F-MAP

99999                                       • package on cards
```

Example 4-3
Subroutine FLEXIN--Manipulation and Data Storage

rewound after the data for the package has been read in.*

It is necessary for the subroutine to know which package it is to manipulate and what set of statements should be used for the manipulations. The variable IFORM is used for this purpose. An "assigned go to" statement of the form

GO TO (1, 2), IFORM

is used in Example 4-3 to tell the computer what instructions in the subroutine should be used for a particular package.** If IFORM = 1, then the computer will jump to the first statement number contained in the parenthesis (statement number 1); if IFORM = 2, it will go to the second number in parentheses (statement number 2). The set of instructions to be used when IFORM = 1 is contained between the "1 CONTINUE" statement and the "RETURN" statement associated with it. In other words, when IFORM = 1, the following instructions will be performed:

```
    1       CONTINUE
            IF (FIRST) REWIND 12
            READ (5,10) A,B,C
   10       FORMAT (3F10.0)
            WRITE (12) C
            T = A/B
            RETURN
```

---

* Any non-blank character in columns 11-15 will cause unit 12 to be rewound after the package is read in -- "x" in column 13 is used by convention. The SYMAP program generally expects the data set to be written and read on logical unit 12; if another unit is used, this rewind command has no useful purpose.

**A maximum of twenty-five sets ofinstructions may be used in any one submission. Not every set of instructions need be referred to in a particular run, but every package to be manipulated must have a set of instructions associated with it, generally by means of the "assigned go to" statement. IFORM is generally used to pass the identifying number from the package to the subroutine, but it may also be used to return the number 99999 from the subroutine to notify the program that the package has been entirely read in.

These have the following meaning or function:

--- IF(FIRST) REWIND 12: if this is the first time the subroutine is called by the E-VALUES package, then the tape or disk file mounted on logical unit 12 is rewound to the beginning; in this way the first value in the data set will be read the first time the subroutine is called. The statement is ignored the remaining nine times the subroutine is called by the E-VALUES package.

--- READ(5,10) A, B, C: for each of the ten times the subroutine is called, the values for the three variables A, B, and C are read in by the card reader (unit 5) according to the format specified in statement number 10.

--- 10 FORMAT (3F10.0): the variables A, B, and C are floating point numbers; each has 10 columns reserved for it, with no digits after the decimal.

--- WRITE (12) C: the value for the variable C is written out -- without a format -- onto the tape or disk file on unit 12 each time the subroutine is called. A formatted write statement of the form

```
          WRITE (12,200) C
    200   FORMAT (F10.0)
```

could have been used, although the unformatted one is generally easier to specify and more efficient for the computer. By this procedure the 10 values of C are read in from the data bank and stored on a temporary data set for subsequent use with the second map.

--- T = A/B: the values to be mapped are returned by the variable T; each value is determined within the subroutine by dividing the value for A by the corresponding value for B. This ability to create new variables within the subroutine greatly adds to the flexibility for the user.

The set of instructions to be used when IFORM = 2 is contained between the "2 CONTINUE" and corresponding "RETURN" statements. The first time the subroutine is called, when the second E-VALUES package is encountered by the program, the tape or disk file on unit 12 will be rewound to the beginning. Then the ten values for the variable C are read, one each time the subroutine is called. The variable T is assigned each value of C so as to transfer the appropriate values for the second map back to the main program.

It is essential that the B-DATA POINTS and E-VALUES packages have a 1:1 correspondence so that the correct value will be associated with each data point. In many cases this can be facilitated by the use of a data bank which contains the x and y coordinates and the associated values on one "record".* Example 4-4 shows the subroutine and packages that could be used to produce the same two maps as in Examples 4-1, 4-2, and 4-3 if the data points and both sets of values are contained in the same data bank.

For IFORM = 1, the data bank is read in, the set of instructions starting with "1 CONTINUE" is used, the x and y coordinates are returned to the main program for mapping, and the two sets of values are written out onto a tape or disk storage file. For IFORM = 2, the values for the first map are read in from the storage file according to the instructions starting with the statement "2 CONTINUE", and the values are returned to the program

---

* As used here a "record" would be the card or cards read in for each point; in Example 4-3 each set of values for the variables A, B, and C is a record. Likewise, the variable C is written out onto tape or disk as one record.

```
                                                         •job control cards
        SUBROUTINE FLEXIN(IFORM,T,FIRST)
        DIMENSION T(2)                                   •subroutine
        LOGICAL FIRST
        GO TO (1,2,3), IFORM
      1 CONTINUE
        IF(FIRST)REWIND 12                               •used with data points
        READ(5,100) T(1),T(2),V1,V2                       package
    100 FORMAT(2F10.1,2F10.0)
        WRITE(12) V1,V2
        RETURN
      2 CONTINUE
        IF(FIRST) REWIND 12                              •used with first values
        READ(12) T(1)                                     package
        RETURN
      3 CONTINUE
        IF(FIRST) REWIND 12                              •used with second values
        READ (12) V1,T(1)                                 package
        RETURN
        END

                                                         •job control cards
A-OUTLINE

                                                         •package on cards
99999
B-DATA POINTS      X
   1   10
                                                         •package on cards;
                                                          read by FLEXIN
99999
C-OTOLEGENDS

                                                         •package on cards
99999
E-VALUES           X
   2   10   X                                            •package on tape or disk
99999                                                     (unit 12) read by FLEXIN
F-MAP

                                                         •package on cards
99999
E-VALUES           X
   3   10   X                                            •package on tape or disk
99999                                                     (unit 12) read by FLEXIN
F-MAP

                                                         •package on cards
99999
```

Example 4-4
Subroutine FLEXIN--Dimensioning and Data Points

for mapping through the variable T. The same type of procedure is used for the second set of values when IFORM = 3.

The variable T is used to return all data to the main program concerning the packages to be used for mapping. In the previous examples, T was used only to return values, but in Example 4-4, it is also used to return the data point locations. Because data points require two variables -- one each to describe the x and y coordinates -- T is "dimensioned" to create two new variables, T(1) and T(2).* For the outline, data points, legends, or barriers packages T(1) is always used for the down or y coordinate and T(2) for the across or x coordinate. Since the variable T has been dimensioned, it should be used throughout as a dimensioned variable. Consequently, when IFORM = 2 or 3, the values should be returned through the variable T(1), rather than T, as in Example 4-3.

When IFORM = 3, both variables are read in; the first value is read in with the variable name "V1" and the second with the name "T(1)". Therefore, the second value is returned to the program to be mapped. When IFORM = 2, only the first variable is read in and it is returned to the main program to be mapped. It is not necessary to read variables on a record past the one to be used; the program will automatically start reading a new record each time the subroutine is called. The following statements would all have the same result for IFORM = 2:

---

* It is necessary to include a DIMENSION statement so that the program will know how many 'T" variables to expect; in this case there are only two, T(1) and T(2). This statement must preceed the first use of the dimensioned variable; the LOGICAL FIRST statement also must preceed the first use of that variable.

```
READ (12)      T(1)
READ (12)      T(1), V2
READ (12)      V1, V2 and   T(1) = V1
READ (12)      T(1), VALTWO.
```

4.3. MANIPULATION OF COORDINATES

In addition to the functions of reading in data for various packages and performing manipulations to produce new variables to be mapped, the subroutine can be used to manipulate or transform the x and y coordinates prior to mapping. For many cases the coordinates will have been provided or measured from a source map, with an upward increasing y axis, whereas SYMAP requires a downward increasing y axis for mapping purposes. In other cases the user may merely wish to change the scale or origin of the coordinate system.

Example 4-5 shows the subroutine and packages required to produce the same two maps as in previous examples when all except the map packages are to be manipulated by the subroutine as follows:

--IFORM = 1: starting with statement number 10, the outline coordinates are read in from cards according to the format in statement 100. The axes are transformed and the origin changed as follows:

$$y = 100 - y$$
$$x = x + 10$$

The x and y coordinate values are then returned to the main program as T(2) and T(1), respectively.

```
                                             · job control cards
      SUBROUTINE FLEXIN(IFORM,T,FIRST)
      DIMENSION T(3)                         · subroutine
      LOGICAL FIRST
      GO TO(10,20,15,30,40),IFORM
   10 CONTINUE                               · used with outline package
      READ(5,100) T(1),T(2)
  100 FORMAT(10X,2F10.0)
   15 CONTINUE                               · used with legends package
      T(1)=100.-T(1)
      T(2)=T(2)+10.0
      RETURN
   20 CONTINUE
      IF(FIRST)REWIND 12                     · used with data points
      READ(5,200)X,Y,Z                         package
      WRITE(12,210)Z
  200 FORMAT(3F10.0)
  210 FORMAT(F10.0)
      T(1)=100.-Y
      T(2)=X+10.0
      RETURN
   30 CONTINUE
      IF(FIRST)REWIND 12                     · used with first values
      READ(12,210) T(1)                        package
      RETURN
   40 CONTINUE
      READ(5,400) (T(I),I=1,3)               · used with second values
  400 FORMAT(10X,F10.0,20X,2A4)                package
      RETURN
      END

                                             · job control cards
A-OUTLINE          X
    1    1
                                             · package on cards;
                                               read by FLEXIN
99999
B-DATA POINTS      X
    2   10
                                             · package on cards;
                                               read by FLEXIN
99999
C-UTOLEGENDS       X
    3
                                             · package on cards;
                                               read by FLEXIN
99999
E-VALUES           X
    4   10  X                                · package on tape or disk;
99999                                          read by FLEXIN
F-MAP

                                             · package on cards
99999
E-VALUES           X
    5   10
                                             · package  on cards
                                               read by FLEXIN
99999
F-MAP                                        · package on cards

99999
```

Example 4-5
Subroutine FLEXIN--Axes Transformation

--IFORM = 2: starting with statement number 20, the data bank of x and y coordinates for the data points and associated 10 values for the first map are read in from cards according to the format in statement number 200. The 10 values are written out onto unit 12, one by one, using the format in statement number 210. Finally, the x and y coordinates are manipulated by the same formulae as were used with the outline coordinates before returning the values to be mapped.

--IFORM = 3: the cards for the C-OTOLEGENDS package are read in automatically by the main program according to the standard format specified in section 2, Preparation of Packages. Then, starting with statement number 15, the x and y coordinates are manipulated by the same statements used for the outline coordinates and returned. Notice that the same RETURN statement is used to end the set of instructions starting with "10 CONTINUE" and the set starting with "15 CONTINUE".

--IFORM = 4: starting with statement number 30, the first set of values is read from the data set on unit 12, using the format in statement number 210, and returned for mapping.

--IFORM = 5: starting with statement number 40, the second set of values is read in from 10 cards, one value per card, according to the format in statement number 400. Notice that there are three variables on each card: T(1), T(2), and T(3). T(1) is the value to be mapped; T(2) and T(3) are used to read in "alphanumeric" identifiers or codes which are then printed out on the output for the appropriate package, preceding the map. For instance, if we are mapping data for 10 towns the identifiers might well be the town names.

Up to eight letters may be used; the format required for T(2) and T(3) is generally 2A4, meaning that the first four letters or numbers are read in or written out by T(2) and the last four by T(3).

In the packages themselves, the second card gives the value for IFORM, the number of times the package is to be called, and the tape or disk file rewind command for the first values package. For the data points and values packages, the number of times the package is to be called equals the number of data points, 10; for the outline package it is the number of islands, 1; and for the legends package the variable is not used.

5. PROGRAMMING AND SYSTEMS GUIDE

This programming and systems guide is intended to provide technical information to people interested in the SYMAP program or who may wish to implement the program at their local computing installation. A knowledge of FORTRAN as well as familiarity with the systems procedures at the local computing installation is assumed.

This guide is for use with SYMAP Version 5 as written for use on the IBM 360 Computer.

5.1. GENERAL INFORMATION

5.1.1. Minimum Machine Configuration

The SYMAP program in its present form requires an IBM 360 computer under Operating System 360 with at least 128K units of storage, a G-level Fortran IV compiler, sufficient auxiliary storage to allocate temporary work areas for the four required execution data sets, and standard input-output facilities including a line printer. In regard to the Fortran Compiler, it is assumed that PGM=G invokes the FORTRAN G compiler. Check with the installation at which you intend to run to verify this.

The required work data sets are FT01F001 through FT04F001, which are used during the processing of any map. Units 5,6, and 7 are needed for card input, the printer and card output. Two other data sets, FT08F001 and FT12F001, may be needed if certain user-specified conditions are true. FT08F001 identifies the unit which receives the data set containing the map location values written when elective 21 of F-MAP package is specified. The SYMAP program is written to accommodate a logical unit 12 when the user Subroutine Flexin is being used with a data bank. The DCB parameters for FT12F001 are therefore user-specified.

The SYMAP program is run at Harvard on an IBM System 360/65, with 256K bytes of internal storage and an additional million bytes of storage provided by an LCS unit, under Operating System release 18, MVT. The unit assignments made in the examples reflect the availability of work space on direct access storage units. This eliminates the need for some of the tape handling that would be necessary if all of the work files were on tape.

5.1.2. Program Storage Requirements

Version 5 of the SYMAP program requires approximately 150K units of storage if implemented with the overlay structure as specified. The program can be run in a 210K partition without overlays.

The storage requirements of the program may be decreased somewhat by reducing the size of the blank COMMON area which is currently specified in the MAIN as having a 10,000 word capacity. The blank COMMON area is used for storage of the B-DATA POINTS, E-VALUES and E1-VALUES INDEX packages and additional information required by specific F-MAP packages. It is also used as the work area for computation of the maximum-sized segment of the map. The program performs its calculations sequentially on individual segments of the map. Decreasing the size of the blank COMMON area imposes stricter limitations on maximum map sizes. It increases the running time because the map is processed in smaller and therefore more numerous segments. Conversely, increasing the size of blank COMMON allows larger maps to be produced; it has the effect of decreasing execution times because the map is processed in larger segments. In fact the effect of enlarging blank COMMON by a factor of one-half or more is so pronounced that, if there is a choice between running the program without overlays or increasing the size of blank COMMON by a factor of 50 percent or more, it is recommended that the size of blank COMMON be increased.

Changes to the size of the blank COMMON area may be effected by changing card E0004550 which reads 'COMMON COMMON(10000)' and card E0010400 which communicates the size of the COMMON area to the program. This is discussed in the section on blank COMMON.

5.1.3. Input and Output Units

The SYMAP program has statements which reference the following FORTRAN IV logical tape/disk data sets: numbers 1, 2, 3, 4, 5, 6, 7, 8 and 12. These units are referenced in four kinds of statements: READ/WRITE (1) X, Y, Z, for binary reads and writes; READ/WRITE (6, 1040) for EBCDIC reads and writes; REWIND 2; and END FILE 3. The units are used as follows:

-- Unit 5 is the system input unit and is only in EBCDIC form;

-- Unit 6 is the system output unit and is only written in EBCDIC form. The maximum record length with carriage control character is 132 characters;

-- Unit 7 is the system output card punch and is written in EBCDIC form.

Units 1-4 are used only for intermediate storage, and are written, end filed, rewound, and read:

-- Unit 1 is used to store the A-OUTLINE package;

-- Unit 2 is used for the A-CONFORMOLINES package;

-- Unit 3 is used for the C-OTOLEGENDS package; and,

-- Unit 4 is used for the C-LEGENDS package.

If any of these four packages are omitted from the program input, the corresponding unit will not be used.

-- Unit 8 is used for binary output only when elective 21 is specified by the input and is not normally used.

-- Unit 12 is used for user-specified input in non-standard form, such as a data bank read in through Subroutine FLEXIN. Unit 12 is rewound by the SYMAP program after the input of a package if the user specifies in the package that this is to be done. The user may reference other units in the FLEXIN subroutine, although he would normally reference only Unit 12 for EBCDIC reads if any units were needed.

5.1.4. SYMAP Source Tape

The source program for SYMAP version 5 is available on tape. This tape is a 360 copy of a tape and includes the source deck and sample data for contour, proximal and conformant maps. These 360 tapes are written at 800 BPI, odd parity, EBCDIC mode with 80 character records blocked in 8000 character blocks--one hundred records to a block.

To implement the program at another computer center, it is necessary to convert the tape to a card deck. The proper fields may be punched for each record by using the IBM utility program IEBPTPCH with the control commands shown in Example 5-1. For more information on IBM-supplied utilities, one should see FORM C28-6586 OS/360 UTILITIES.

5.2. PROGRAMMER INFORMATION

5.2.1. Program Times

To have the program times printed out by the program, it is necessary to change the statement on card E0011600, TTEMP=0.0. This statement should be changed so that the CPU time (not the elapsed time) in minutes is stored in TTEMP. This is generally done by calling some system subroutine but it can be done only in computers that have clocks.

5.2.2. User Subroutine

It should be noted that Subroutine FLEXIN occurs on cards E0016900-E0017200. This subroutine is generally replaced by a user-written subroutine. When this is the case the user-written subroutine should be put in the deck in place of the "dummy subroutine" on these cards.

5.2.3. Blank COMMON

All but one of the statements referring to blank COMMON are exactly as follows: COMMON COMMON(1). This statement allows any location in blank COMMON to be referenced by means of the subscripted variable, COMMON. The variable COMMON is used only in CALL statements as an argument and always with a variable as its subscript. The called SUBROUTINE statement has a corresponding subscripted dummy argument.

Example 5-1
Utility Program
To Punch Source Card from SYMAP Tape

```
         1111111111222222222233333333334444444445
1234567890123456789012345678901234567890123456789
//PUNCH EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=X,UNIT=TAPE9T,VOL=SER=
// DISP=(OLD,KEEP),LABEL=(,BLP),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DEN=2)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *

  PUNCH MAXFLDS=1
  RECORD FIELD=(80)


/*
//
```

NOTE: The tapes on which the program is distributed are unlabeled and the user may specify any six character name desired.

In this way, SYMAP is able to use blank COMMON for several arrays of variable size and to achieve a sort of dynamic storage allocation. It is desirable to use as large a part of storage as possible for the map grid (IMATRIX) because the map is not created all at once. It is created in as many sections as are necessary, depending on the size of the section (determined by the amount of core) and the size of the map.

The size of blank COMMON is set by card E0004550 and in the distributed version is 10000 words. This size is communicated to the program by setting the variable NSTOR in the data statement on card E0010400. Thus by changing the number '10000' on these two cards, it is possible to change the amount of blank COMMON used by the program. This may be desirable if core is exceeded or if there is a considerable portion of core remaining unused.

The desirability of increasing the size of blank COMMON is twofold. In the first place, the B-DATA POINTS and E-VALUES packages are stored there. (The E1-VALUES INDEX package is stored there but this package is rarely used.) These require two locations for each point and one for each value. (Plus one for each index if the E1-VALUES INDEX package is used.) In addition, for contour and proximal maps, there is another location used for a locater-index for each point, and an additional two locations used for the slopes at every point for contour maps. Thus, for a contour map there are 6 words used for every point, including its value; if there are 1000 points, this uses up 6000 words of blank COMMON. On the other hand, a conformant map would only require one word for each conformant figure (or, 3 words if data points are included). This would allow the use of many more conformant zones with a conformant map than data points with a contour map, for the same amount of blank COMMON. With the 4000 words left on blank COMMON for a 1000 data point contour map, the map grid would be set up. Assuming a map 10.1 inches wide, this grid would have 100 columns (the width of the map at 10 columns per inch with 1 column used for the border) and only 40 rows. This means that for every 40 rows of the map (5.0 inches), a new section would be needed. A reduction in the number of sections required to make a map is desirable since units 1, 2, and 3 must be rewound at the beginning of each section. Unit 4 is rewound at the end of each map strip.

5.2.4. Labeled COMMON

Labeled COMMON statements are generally contained only in routines which reference them. For a given labeled COMMON, one will often find different variable names assigned to the same location in different routines; this inconsistency may be removed from the program at a later date. The sizes of all the references to a given labeled COMMON are all the same.

5.2.5. Entry Statements

There are a number of ENTRY statements in the program which are used primarily for the transfer of parameters. It is not too difficult to take these statements out of the program, if some care is exercised in doing so.

5.2.6. Checking for Missing Data

In the past Elective 18 of the F-MAP package was used to check for missing values in the data, coded as minus zero's. The following condition was used to test the value of VALUE to see whether it was a '-0.0'. IF (VALUE .EQ. 0.0 .AND. SIGN(1.0,VALUE). LT. 0.0)....... This test worked on the IBM 7094 but it is very possible that it will not work on other computers including the IBM 360. With the FORTRAN language, the result of the SIGN function is undefined when the second argument is equal to 0.0. (It should be noted that -0.0 is equal to 0.0, although a different result is obtained from the SIGN function on the IBM 7094)

There are other difficulties encountered in using Elective 18 to eliminate missing data. It is first necessary to have the data itself assigned the value of -0.0, rather than 0.0. At the Harvard Computing Center the systems routines on the IBM 360 are such that when a blank field is read with an F format, the resulting value is 0.0 rather than -0.0 which was the case with the IBM 7094. To get around this limitation it may be better to use Electives 19 and 20 in many cases, rather than 18; these are standard FORTRAN routines and will function successfully at any installation. As a final note, 360 users should not be confused by the 360's inability to store a -0; this is quite a different thing from a -0.0 which it can store.

5.3. SYSTEM INFORMATION

5.3.1. Overlay Structure

The program is overlayed to conserve core storage. In some machines it may be possible to run the entire program without overlaying, although this extra space can probably be used more efficiently in other ways. It is extremely dangerous to overlay the program further than the present version, particularly with regard to those routines which have more than one entry (by means of the ENTRY statement).

The uses of the overlay links are as follows:

-- Link 0 (the main link and not really an overlay) is the main program, together with a number of subroutines used by different overlays and a large portion of the labeled COMMONs;

-- Link 1 is the input link and its main routine is INPUT. This link should contain subroutine FLEXIN. Note that FLEXIN will not preserve any variables created during input of the previous map. This routine is overlayed after it reads in an F-MAP package whenever a map is made;

-- Link 2 is the map initialization link and its main routine is INIT. Its parent link is 0, and it replaces link 1 at the beginning of a map in order to perform various functions;

-- Link 3 has parent link 0 and overlays link 2 in order to create the actual content of the map. Its main routine is MAP. It is an ancestor of all the secondary and tertiary links;

-- Link 4 is used solely for producing conformant maps and its main routine is FLAT. Parent link is 3 and it has no daughters;

-- Link 5 is a set of subroutines used by links 6 and 7 for making contour and proximal maps. It is called in over link 4 if a contour or proximal map is to be made. Its daughters are the tertiary links 6 and 7;

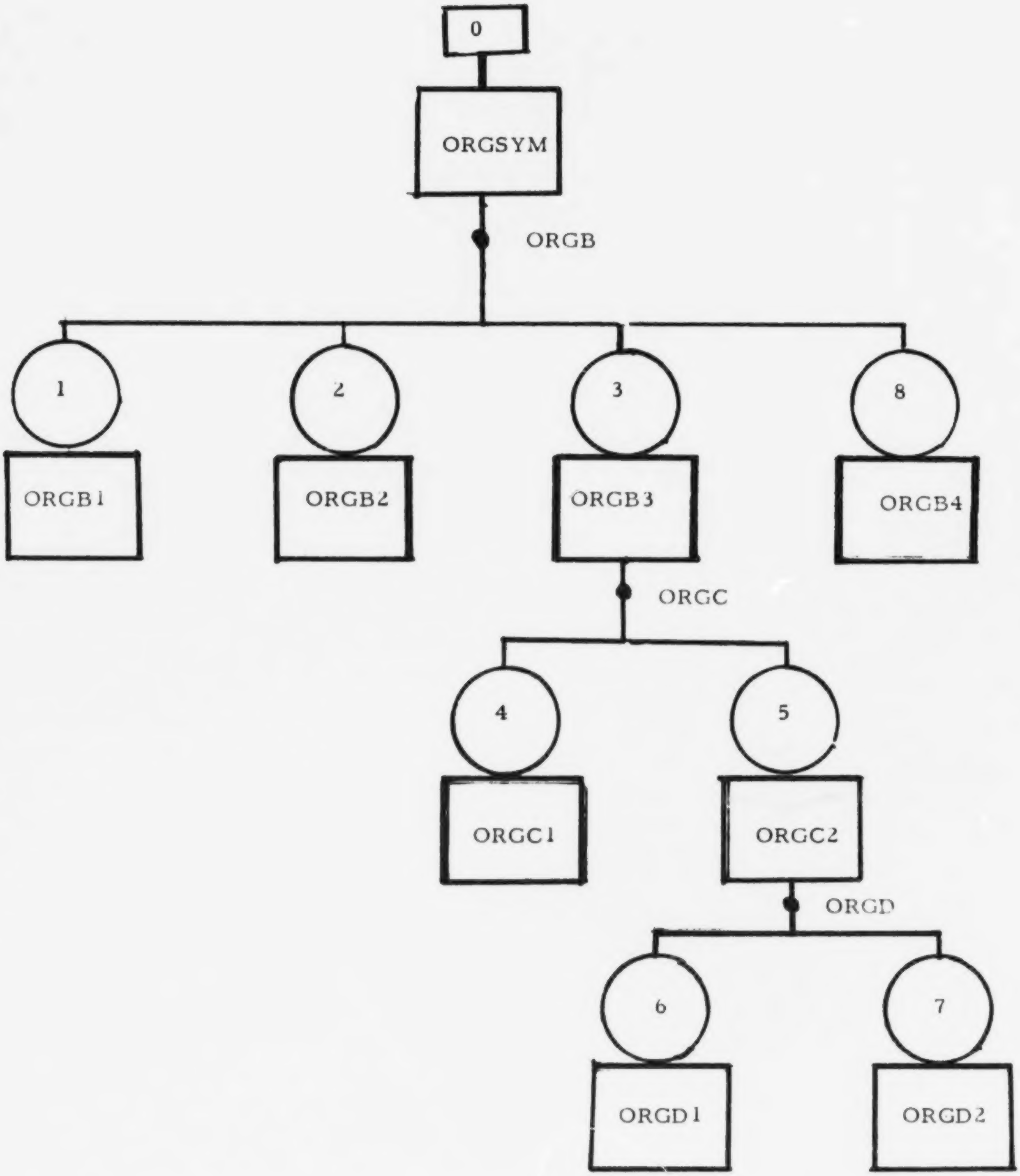
-- Link 6 is used for initialization of contour and proximal maps. Its main routine is SLOPER and its parent link is 5;

-- Link 7 is used for the actual creation of the contour map. Its main routine is CONTOR and its parent link is 5; and,

-- Link 8 replaces link 3 and its daughters and produces the information at the bottom of the map and the histogram. Its main routine is FINAL and its parent link is 0.

The overlay structure is shown in Example 5-2.

Example 5-2
OVERLAY TREE


0
ORGSYM
ORGB
1
ORGB1
2
ORGB2
3
ORGB3
8
ORGB4
ORGC
4
ORGC1
5
ORGC2
ORGD
6
ORGD1
7
ORGD2

5.3.2. Generation of Overlays

For ease of use and economy in running time, it is desirable to keep load modules for each program segment stored on disk. This makes it easier to incorporate program changes and to use certain sections of the program for special testing. Programmers may wish to change various routines without altering the original program modules.

For example, when the FLEXIN subroutine is being used to read input, the subroutine alone can be compiled and link-edited into the program.

Subroutine FLEXIN--otherwise known as subroutine 15--may be used to specify data formats which differ from those used by the packages as standard. It may also be used to perform various arithmetic operations on the data to be mapped. This subroutine is one of the most important features of the program. It provides the user with complete flexibility of data inputs and data manipulation. However, since the subroutine is usually changed for each submission of the program, it is necessary to compile this subroutine each time the program is run.

This procedure requires the generation of a library data set which has each of the program segments stored as a separate load module. The user will have to refer this problem to the systems programmers at his installation in order to obtain permanent residence space. Auxiliary control statements may be necessary for the successful completion of this and succeeding steps. To do this requires use of the editing facilities of the OS Linkage Editor (refer to FORM C-28-6538 IBM SYSTEM/360

Operating System Linkage Editor for a full description of the linkage editor facilities). The linkage editor generates a separate load module for each name segment which it encounters. Proper placement of name statements and specification of the 'NCAL' option to the linkage editor will create a data set which contains each segment as a member in load module form.

Example 5-3 shows the control cards necessary to set up the main data set when the input to the linkage editor is to be in the form of binary decks. The controls to execute the linkage editor program have been completely written out in this job since it is not standard to have a catalogued procedure for FORTRAN which only does a link edit. The user should note that the data set name and the volume identification (volid) must be specified in the SYSLMOD DD statement before this program can be run. Note also that the NAME statements in the linkage editor input refer to the object decks which appear before them; in other words, when the linkage editor encounters a NAME statement, it forms a load module of the object decks which it had encountered up to that point and starts another load module for the object decks which follow.

Example 5-4 shows the control card structure required to compile the nine segments of SYMAP from the source tape and then to link edit each segment into a load module.

5.3.3. Running Efficiency

It is foolish to run the source decks of the SYMAP program more than once. Binary decks should be obtained immediately and each source deck replaced with the corresponding binary deck (object deck). Subroutine FLEXIN will generally be submitted as a source deck, however.

Example 5-3
Control Cards to Link Edit Object Modules Into Load Modules as the Overlay Segments

```
         111111111122222222223333333333444444444455555555556666666666777
123456789012345678901234567890123456789012345678901234567890123456789012
//LKEC      EXEC PGM=IEWL,PARM=(XREF,LIST,LET,NCAL),TIME=(7,0),            X
//               REGION=150K
//SYSPRINT DD  SYSOUT=A
//SYSLIB DD  DSNAME=SYS1.FORTLIB,DISP=SHR
//SYSLIN DD  DDNAME=SYSIN
//SYSLMOD DD DSN=A432701.SYM.B,UNIT=SYSLIB,
// VOL=(PRIVATE,SER=LCGSA1),DISP=(NEW,KEEP),
// SPACE=(TRK,(30,,2))
//SYSUT1 DD UNIT=SYSQUE,SPACE=(TRK,(30,,2)),SEP=(SYSLIN,SYSLMOD),
//               DSNAME=&&DISKA
//SYSIN DD  *
```

Insert object decks for SYMAPV5 (main deck) and the following subroutines: TIMER, BLOCK DATA, VALID, LEVEL.

```
 NAME ORGSYM
```

Insert object decks for: FLEXIN, INPUT, INVALS, INPTS, ININD, INLINS, INFLAT, INCOVS, INLEGS, NUMCHAR, INBARS, INMAP.

```
 NAME ORGB1
```

Insert object decks for: INIT, SHEPIN, DIMSET, INDXER, SORT, LEVSET, LIMIT, PROGIF, STAND, PRDPTS.

```
 NAME ORGB2
```

Insert object decks for: MAP, BORDER, SECTIN, LINE, OUT, LEGD, ALINE, INSECT, GON, COVER, DATSYM.

```
 NAME ORGB3
```

Insert object decks for: FLAT, BANDR.

```
 NAME ORGC1
```

Insert object decks for: ICLOSE, BARIN.

```
 NAME ORGC2
```

Insert object deck for: SLOPER.

```
 NAME ORGD1
```

Insert object decks for: CONTOR, INFLSM, ININTP, BLANK.

```
 NAME ORGD2
```

Insert object decks for: FINAL, HGRAM.

```
 NAME ORGB4
/*        OS END OF FILE CARD
```

Example 5-4
Compile and Link Edit SYMAP Into Overlay Segment Load Modules

```
         111111111122222222223333333333444444444455555555556666666666777
123456789012345678901234567890123456789012345678901234567890123456789012

//ALLOC EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=A
//DD1           DD DSN=A432701.SYM.B,DISP=(NEW,KEEP),UNIT=SYSLIB,
//  VOL=(PRIVATE,SER=LCGSA1),SPACE=(TRK,(30,,2))
/*
//SETUP  EXEC  PGM=IEBGENER,PARM=MOD,REGION=150K
//SYSPRINT  DD  SYSOUT=A
//SYSUT1 DD DSN=MOD1,DISP=(OLD,KEEP),VOL=SER=HU1292,UNIT=TAPE9T,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DEN=2),LABEL=(,NL)
//SYSUT2 DD DSN=CLLIB,UNIT=SYSQUE,DISP=(NEW,PASS),                     X
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),                     X
//  SPACE=(TRK,(80,10,10))
//SYSIN  DD  *,DCB=BLKSIZE=80
      GENERATE MAXNAME=11,MAXGPS=11,MAXFLDS=11
      MEMBER NAME=(CRGSYM)
      RECORD IDENT=(8,'E0016700',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGB1)
      RECORD IDENT=(8,'E0184000',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGB2)
      RECORD IDENT=(8,'E0257800',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGB3)
      RECORD IDENT=(8,'E0317100',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGC1)
      RECORD IDENT=(8,'E0328900',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGC2)
      RECORD IDENT=(8,'E0388000',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGD1)
      RECORD IDENT=(8,'E0395000',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGD2)
      RECORD IDENT=(8,'E0439400',73),FIELD=(80,1,,1)
      MEMBER NAME=(CRGB4)
      RECORD IDENT=(8,'E0465500',73),FIELD=(80,1,,1)
```

```
/*
//C1   EXEC  FORTHC,PARM.FORT='NAME=SYMAP,LIST,DECK',LOADSET=C1
//FORT.SYSIN  DD DSN=CLLIB(ORGSYM),UNIT=SYSQUE,DISP=(CLD,PASS)
/*
//C2 EXEC FORTHC,PARM.FORT='OPT=1,LIST,DECK',LOADSET=C2,
// REGION.FORT=350K
//FORT.SYSIN DD DSN=CLLIB(ORGB1),UNIT=SYSQUE,DISP=(OLD,PASS)
/*
//C3 EXEC FORTHC,PARM.FORT='LIST,DECK',LOADSET=C3
//FORT.SYSIN DD DSN=CLLIB(ORGB2),UNIT=SYSQUE,DISP=(OLD,PASS)
/*
//C4 EXEC FORTHC,PARM.FORT='LIST,DECK',LOADSET=C4
//FORT.SYSIN DD DSN=CLLIB(ORGB3),UNIT=SYSQUE,DISP=(CLD,PASS)
/*
//C5 EXEC FORTHC,PARM.FORT='LIST,DECK',LCADSET=C5
//FORT.SYSIN DD DSN=CLLIB(ORGC1),UNIT=SYSQUE,DISP=(OLD,PASS)
/*
//C6 EXEC FORTHC,PARM.FORT='LIST,DECK',LOADSET=C6,REGICN.FORT=350K
//FORT.SYSIN DD DSN=CLLIB(ORGC2),UNIT=SYSQUE,DISP=(CLD,PASS)
/*
//C7 EXEC FORTHC,PARM.FORT='LIST,DECK',LCADSET=C7
//FORT.SYSIN DD DSN=CLLIB(ORGD1),UNIT=SYSQUE,DISP=(OLD,PASS)
/*

//C8 EXEC FORTHC,PARM.FORT='LIST,DECK',LOADSET=C8,REGICN.FCRT=350K
//FORT.SYSIN DD DSN=CLLIB(ORGD2),UNIT=SYSQUE,DISP=(OLD,PASS)
/*
//C9 EXEC FORTHC,PARM.FORT='LIST,DECK',LOADSET=C9,REGICN.FCRT=350K
//FORT.SYSIN DD DSN=CLLIB(ORGB4),UNIT=SYSQUE,DISP=(CLD,PASS)
/*
//LINK  EXEC  LKED,PARM.LKED=(XREF,LET,LIST,NCAL)
//LKED.SYSLMOD DD DSN=A432701.SYM.B,DISP=(OLD,KEEP),UNIT=SYSLIB,
//               VOL=(PRIVATE,SER=LCGSA1),SPACE=(1024,(400,200,2))
//DD1   DD    DSN=C1,UNIT=SYSQUE,DISP=CLD
//DD2   DD    DSN=C2,UNIT=SYSQUE,DISP=CLD
//DD3   DD    DSN=C3,UNIT=SYSQUE,DISP=CLD
//DD4   DD    DSN=C4,UNIT=SYSQUE,DISP=CLD
//DD5   DD    DSN=C5,UNIT=SYSQUE,DISP=CLD
//DD6   DD    DSN=C6,UNIT=SYSQUE,DISP=CLD
//DD7   DD    DSN=C7,UNIT=SYSQUE,DISP=CLD
//DD8   DD    DSN=C8,UNIT=SYSQUE,DISP=CLD
//DD9   DD    DSN=C9,UNIT=SYSQUE,DISP=CLD
//LKED.SYSIN  DD   *
```

Example 5-4
(continued)

Example 5-4
(Continued)

```
         INCLUDE DD1
         NAME ORGSYM
         INCLUDE DD2
         NAME ORGB1
         INCLUDE DD3
         NAME ORGB2
         INCLUDE DD4
         NAME ORGB3
         INCLUDE DD5
         NAME ORGC1
         INCLUDE DD6
         NAME ORGC2
         INCLUDE DD7
         NAME ORGD1
         INCLUDE DD8
         NAME ORGD2
         INCLUDE DD9
         NAME ORGB4
/*            OS END OF FILE CARD
```

NOTE: LOADSET refers to the dsname of SYSLIN in the compile steps. These loadset names are used as input to the linkage editor.

The loading time (the time the linkage editor uses to load systems programs and the SYMAP program) is approximately four seconds at the LCG. There is a significant efficiency achieved by running several maps at once. Example 5-5 shows a SYMAP submission as made at the LCG. This listing includes the job control language that is brought in with the catalogued procedures SYMCLG and SYMLINK1.

Since the OS job control language is tedious, it may be time saving to set up the control statements necessary to execute the program, with and without changes to the user subroutine, as catalogued procedures. This is particularly useful when a large number of maps will be produced. Updates to the various routines may be made by compiling the appropriate control sections and link-editing them into the load module which contains the proper segment. Basically the same method as is used for the conversion of the user subroutine can be employed here. It is recommended that the user familiarize himself with the IBM Operating System Linkage Editor publication before attempting this operation.

5.3.4. Program Execution

To execute the program, the separate load modules which contain the program segments must be re-link-edited and the overlay structure called in. The basic control cards used at Harvard to load and execute the program with the correct overlay structure are shown in Example 5-6. In order to change SUBROUTINE FLEXIN--or any other subroutine the user desires--one compile step is needed for each program segment containing routines to be changed. Each of the compilation steps is given a step name and a load set name. The SYSLIN data set must

Example 5-5
SYMAP submission at the LCG

```
          111111111122222222223333333333444444444455555555556666666666777777777778
12345678901234567890123456789012345678901234567890123456789012345678901234567890
// EXEC SYMCLG,LEVEL=B
XX PROC LEVEL=A,L=NL,PRINT=6,CARD=[illegible],TAPE=SCRTCH,BPI=2,BLOCK=1572*,  00000000
XX REG=150                                                                    00000010
XXFORT    EXEC PGM=G,REGION=150K                                              00000020
XXSYSPRINT DD  SYSOUT=A,DCB=(LRECL=120,RECFM=FBA,BLKSIZE=7200)                00000030
XXSYSPUNCH DD  SYSOUT=B,SPACE=(80,(200,50),RLSE)         *950 CARD MAX*       00000040
XXSYSLIN    DD DSNAME=FLEX,UNIT=SYSQUE,SPACE=(80,(200,50),RLSE),             X00000050
XX             DISP=(,PASS),DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200)              00000060
//FORT.SYSIN DD *

   0001                SUBROUTINE FLEXIN
   0002                RETURN
   0003                END

XXLKED EXEC PGM=IEWL,PARM=(LET,LIST,OVLY),REGION=&REG,TIME=(0,15)             00000070
XXSYSPRINT DD SYSOUT=A,DCB=(LRECL=121,RECFM=FBM,BLKSIZE=3509)                 00000080
XXSYSLIB    DD DSNAME=SYS1.FORTLIB,DISP=SHR                                   00000090
XXSYMLIB    DD DSNAME=A[illegible]405.SYM.&LEVEL,DISP=SHR                     00000100
XXSYSLMOD   DD DSNAME=&&GOSET(MAIN),UNIT=SYSQUE,DISP=(,PASS),                X00000110
XX             SPACE=(1024,(200,20,1),RLSE)                                   00000120
XXSYSUT1    DD DSNAME=&&DISKA,UNIT=SYSQUE,SPACE=(1024,(200,10)),             X00000130
XX             SEP=SYSLMOD)                                                   00000140
XXSYSLIN    DD DDNAME=CATIN                                                   00000150
XX001       DD DSNAME=FLEX,UNIT=SYSQUE,DISP=(OLD,DELETE)                      00000160
XXCATIN  DD  DISP=SHR,DSN=SYS1.PROCLIB(SYMLINK1)                              00000170
IEW0000                INCLUDE SYMLIB(ORGSYM)                                         E0001800
IEW0000                OVERLAY [illegible]                                            E0001900
IEW0000                INCLUDE [illegible]                                            E0002000
IEW0000                INCLUDE SYMLIB([illegible])                                    E0002100
IEW0000                OVERLAY [illegible]                                            E0002200
IEW0000                INCLUDE SYMLIB([illegible])                                    E0002300
IEW0000                OVERLAY [illegible]                                            E0002400
IEW0000                INCLUDE SYMLIB([illegible])                                    E0002500
IEW0000                OVERLAY [illegible]                                            E0002600
```

Example 5-5
(Continued)

```
IEW0000                 INCLUDE SYMLIB(CFGC1)                                             E0002700
IEW0000                 OVERLAY CFGC                                                      F0002800
IEW0000                 INCLUDE SYMLIB(CFGC2)                                             F0002900
IEW0000                 OVERLAY CFGD                                                      F0003000
IEW0000                 INCLUDE SYMLIB(CFGD1)                                             E0003100
IEW0000                 OVERLAY CFGD                                                      F0003200
IEW0000                 INCLUDE SYMLIB(CFGD2)                                             F0003300
IEW0000                 OVERLAY CFGB                                                      E0003400
IEW0000                 INCLUDE SYMLIB(CFGB4)                                             F0003500
IEW0000                 ENTRY SYMAP                                                       F0003600
XXGO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT),REGION=&REG                         00000180
XXFT01F001 DD DSNAME=&&SYMAP1,UNIT=SYSQUE,SPACE=(TRK,(20,10)),              X00000190
XX DISP=(,DELETE),DCB=(RECFM=VSB,LRECL=84,BLKSIZE=1684,BUFNO=1)              00000200
XXFT02F001 DD DSNAME=&&SYMAP2,UNIT=SYSQUE,SPACE=(TRK,(20,10)),              X00000210
XX             DISP=(,DELETE),DCB=*.FT01F001                                 00000220
XXFT03F001 DD DSNAME=&&SYMAP3,UNIT=SYSQUE,SPACE=(TRK,(20,10)),              X00000230
XX             DISP=(,DELETE),DCB=*.FT01F001                                 00000240
XXFT04F001 DD DSNAME=&&SYMAP4,UNIT=SYSQUE,SPACE=(TRK,(20,10)),              X00000250
XX             DISP=(,DELETE),DCB=*.FT01F001                                 00000260
XXFT05F001 DD DDNAME=SYSIN                                                   00000270
XXFT06F001 DD SYSOUT=(&PRINT,,&CARP),DCB=(RECFM=FBA,LRECL=133,              X00000280
XX BLKSIZE=7250,BUFNO=1)                                                     00000290
XXFT07F001 DD SYSOUT=B,SPACE=(80,(33,117),RLSE,CONTIG),DCB=(RECFM=FB,       00000300
XX LRECL=80,BLKSIZE=3200)                                                    00000310
XXFT08F001 DD DSNAME=F001,UNIT=(2400,,DEFER),DISP=(,PASS),                  X00000320
XX VOL=(,RETAIN,SER=&TAPE),LABEL=(1,&L),DCB=(RECFM=VSB,                      00000330
XX             LRECL=524,BLKSIZE=&BLOCK,EROPT=ACC,DEN=&BPI,BUFNO=1)          00000340
XXFT08F002 DD DSNAME=F002,LABEL=(2,&L),DISP=(,PASS),                        X00000350
XX             VOLUME=REF=*.FT08F001,DCB=*.FT08F001                          00000360
XXFT08F003 DD DSNAME=F003,LABEL=(3,&L),DISP=(,PASS),                        X00000370
XX             VOLUME=REF=*.FT08F002,DCB=*.FT08F001                          00000380
XXFT08F004 DD DSNAME=F004,LABEL=(4,&L),DISP=(,PASS),                        X00000390
XX             VOLUME=REF=*.FT08F003,DCB=*.FT08F001                          00000400
XXFT08F005 DD DSNAME=F005,LABEL=(5,&L),DISP=(,PASS),                        X00000410
XX             VOLUME=REF=*.FT08F004,DCB=*.FT08F001                          00000420
XXFT08F006 DD DSNAME=F006,LABEL=(6,&L),DISP=(,PASS),                        X00000430
XX             VOLUME=REF=*.FT08F005,DCB=*.FT08F001                          00000440
XXFT08F007 DD DSNAME=F007,LABEL=(7,&L),DISP=(,PASS),                        X00000450
XX             VOLUME=REF=*.FT08F006,DCB=*.FT08F001                          00000460
XXFT08F008 DD DSNAME=F008,LABEL=(8,&L),DISP=(,PASS),                        X00000470
XX             VOLUME=REF=*.FT08F007,DCB=*.FT08F001                          00000480
XXFT08F009 DD DSNAME=F009,LABEL=(9,&L),DISP=(,PASS),                        X00000490
XX             VOLUME=REF=*.FT08F008,DCB=*.FT08F001                          00000500
XXFT12F001 DD DUMMY                                                          00000520
//GO.SYSIN DD *
```

Example 5-6
Job Control Language to Load and Execute SYMAP

```
         1111111111222222222233333333334444444444555555555566666666667
1234567890123456789012345678901234567890123456789012345678901234567890123
___________________________________________________________________________
//LKED EXEC PGM=IEWL,PARM=(LET,LIST,OVLY),REGION=150K,TIME=(0,15)
//SYSPRINT DD SYSOUT=A,DCB=(LRECL=121,RECFM=FBM,BLKSIZE=3509)
//SYSLIB   DD DSNAME=SYS1.FORTLIB,DISP=SHR
//SYMLIB DD DSN=A432701.SYM.B,DISP=SHR
//SYSLMOD  DD DSNAME=&&GOSET(MAIN),UNIT=SYSQUE,DISP=(,PASS),           X
//             SPACE=(1024,(200,20,1),RLSE)
//SYSUT1   DD DSNAME=&&DISKA,UNIT=SYSQUE,SPACE=(1024,(200,10)),        X
//             SEP=SYSLMOD
//SYSLIN DD DDNAME=SYSIN
//SYSIN DD *
          INCLUDE SYMLIB(ORGSYM)
          OVERLAY ORGB
          INCLUDE SYMLIB(ORGB1)
          OVERLAY ORGB
          INCLUDE SYMLIB(ORGB2)
          OVERLAY ORGB
          INCLUDE SYMLIB(ORGB3)
          OVERLAY ORGC
          INCLUDE SYMLIB(ORGC1)
          OVERLAY ORGC
          INCLUDE SYMLIB(ORGC2)
          OVERLAY ORGD
          INCLUDE SYMLIB(ORGD1)
          OVERLAY ORGD
          INCLUDE SYMLIB(ORGD2)
          OVERLAY ORGB
          INCLUDE SYMLIB(ORGB4)
          ENTRY SYMAP
/*
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT),REGION=150K
//FT01F001 DD DSNAME=&&COMAP1,UNIT=SYSQUE,SPACE=(TRK,(20,10)),         X
// DISP=(,DELETE),DCB=(RECFM=VSB,LRECL=84,BLKSIZE=1684,BUFNO=1)
//FT02F001 DD DSNAME=&&COMAP2,UNIT=SYSQUE,SPACE=(TRK,(20,10)),         X
//             DISP=(,DELETE),DCB=*.FT01F001
//FT03F001 DD DSNAME=&&COMAP3,UNIT=SYSQUE,SPACE=(TRK,(20,10)),         X
//             DISP=(,DELETE),DCB=*.FT01F001
//FT04F001 DD DSNAME=&&COMAP4,UNIT=SYSQUF,SPACE=(TRK,(20,10)),         X
//             DISP=(,DELETE),DCB=*.FT01F001
//FT05F001 DD DDNAME=SYSIN
//FT06F001 DD SYSOUT=(W,,9901),DCB=(RECFM=FBA,LRECL=132,
// BLKSIZE=7260,BUFNO=1)
//FT07F001 DD SYSOUT=B,SPACE=(80,(39,117),RLSE,CONTIG),DCB=(RECFM=FB,
// LRECL=80,BLKSIZE=3200)
//FT08F001 DD DSN=F001,UNIT=(2400,,DEFER),DISP=(,PASS),
// VOL=(,RETAIN,SER=VOLID1),LABEL=(1,NL),DCB=(RECFM=VSB,
// LRECL=524,BLKSIZE=15724,EROPT=ACC,DEN=2,BUFNO=1)
//FT12F001 DD DUMMY
//SYSIN DD *
```

```
/*
```

Example 5-7
Job Control Language to Override
SYMAP subroutines with edited subroutines

```
         11111111112222222222333333333344444444445555555555666666666677
12345678901234567890123456789012345678901234567890123456789012345678901
```

```
//S1    EXEC FORTGC,LOADSET=S1,PARM.FORT='DECK,LIST'
//FORT.SYSIN DD *
```

Altered Source Deck of FLEXIN, INPUT, INVALS

```
/*
//S2    EXEC FORTGC,LOADSET=S2,PARM.FORT=(DECK,LIST)
//FORT.SYSIN DD *
```

Altered Source Decks of LEGD and DATSYM

```
/*
// EXEC SYMCLG,LEVEL=B,PARM.LKED=(LET,OVLY,XREF,LIST,MAP),REG=150K
//FORT.SYSIN  DD *
      SUBROUTINE FLEXIN
      RETURN
      END
/*          OS END OF FILE CARD
//LKED.SYSLIN  DD   DDNAME=SYSIN
//S1 DD DSNAME=S1,DISP=(OLD,DELETE)
//S2    DD  DSNAME=S2,DISP=(OLD,DELETE)
/*
//LKED.SYSIN DD *
  INCLUDE SYMLIB(ORGSYM)
  OVERLAY ORGB
  INCLUDE S1
  INCLUDE DD1
  INCLUDE SYMLIB(ORGB1)
```

Example 5-7
(Continued)

```
    OVERLAY  ORGB
    INCLUDE SYMLIB (ORGB2)
    OVERLAY ORGB
    INCLUDE S2
    INCLUDE  SYMLIB(ORGB3)
    OVERLAY ORGC
    INCLUDE  SYMLIB(ORGC1)
    OVERLAY ORGC
    INCLUDE  SYMLIB(ORGC2)
    OVERLAY ORGD
    INCLUDE SYMLIB (ORGD1)
    OVERLAY ORGD
    INCLUDE SYMLIB(ORGD2)
    OVERLAY ORGB
    INCLUDE SYMLIB(ORGB4)
    ENTRY SYMAP
/*
//GO.SYSIN DD *,DCB=BUFNO=1


/*          OS END OF FILE CARD
```

This example shows a program running with changes to FLEXIN, INPUT, INVALS of segment two and LEGD and DATSYM of segment four.

It assumes that SYMLIB defines the data set which contains the previously link-edited program segments of SYMAP, with the naming conventions established in the previous examples.

LOADSET refers to the dsname of SYSLIN in the compile steps. These loadset names are used as input to the linkage editor.

have either a keep or pass status. Then the changes may be specified to the linkage editor by including the data sets output by the compiler. This output should be link-edited into the appropriate overlays before the inclusion of the data set which specifies the load module. (See Example 5-7)

5.4. PRINTER INFORMATION

5.4.1. Spacing of Rows of Output

Most line printers print at six lines per inch. A number of computer facilities do not have the ability to print 8.0 lines per inch, the standard used for SYMAP. There is an elective in the program to change the line spacing, but if the standard is to be other than 8.0, it is desirable to change the standard in the program. This can be done by changing the 8.0 on card E0209100 to 6.0 or whatever number is desired.

5.4.2. Carriage Control Characters for Overprinting

To be able to use overprinted symbols on the map the computer which does the printing must have an instruction for overprinting. There is an elective to change the method of overprinting if it is done differently than on the IBM 360. On the 360 a blank control character is interpreted as "space and then write", and control character '+' indicates "do not space before writing". The first line has a blank control character and the following three a '+' control character. Elective 26 of the F-MAP package can be used to specify a non-standard method of overprinting: three lines with control character '+', indicating "write and do not space", followed by one line with a blank control character, indicating "write and space".

The other carriage control characters employed are a '1' for "skip to a new page", and '0' for "skip a line".

5.4.3. Printer Carriage Control Tape

A special carriage control tape must generally be used for printing SYMAP output, particularly in the case of maps which are greater than one page in length. With a standard carriage tape, the printer will leave a gap at the bottom of every page--this will leave gaps in the map. Printers which have the option of running at 6.0 or 8.0 lines per inch will fail to skip to the top of a page if a standard 6-line-per-inch carriage tape is used for printing 8-line-per-inch output. At Harvard a "special SYMAP carriage tape" is requested as part of the job control language -- the number of the carriage control tape is 9901. This carriage tape is set up so that ejection places the paper at the top of a new page (88 lines for the entire tape) when printing at 8 lines per inch; the paper is not ejected at the bottom of a page (no punch in the 9-channel on most machines). A different tape would have to be used for printing at other than 8 lines per inch.

6. THE "SYMVU" PROGRAM

6.1. INTRODUCTION

SYMVU is a computer graphics program written for the purpose of generating three dimensional line-drawing displays of data. The SYMVU program is usable by persons with very little programming experience. Only three control cards are necessary for the generation of the graphic displays. Thirty-two electives or options are built into the program allowing for considerable flexibility in generating the displays of data. However, only seven of these options are absolutely necessary for the production of a single display.

SYMVU is most commonly used for quantitative geographic mapping purposes. Unlike contour mapping, SYMVU illustrates the absolute values of spatially continuous data. Contour maps (or isoline maps) use rounded intervals for mapping displays of quantitative information. In the case of SYMVU absolute values of a continuous surface are illustrated. The SYMVU program also has the capabilities of conformant and proximal mapping using data generated by the SYMAP program.

One of the distinct features of this program is that it accomplishes the task of deciding which parts of the object being viewed are seen and which are hidden from view.

The program is written in FORTRAN-IV and is currently being operated on the IBM 360-65 using 220K memory. The output device that draws the illustrations is a Cal-Comp plotter. Runs have been successfully completed on both an 11 inch and 30 inch plotter, although, because of a number of size constraints within the program, a 30 inch plotter presently has little advantage.

6.1.1. Basic Principles

The program utilizes data which is in the form of a grid matrix. This matrix information can be generated in either of two ways.

The first and most common alternative is to utilize data generated by the SYMAP program as a two dimensional graphic display. SYMAP interpolates data from point or area locations so as to produce spatially continuous surfaces organized in the form of a grid matrix. This data can then be utilized by the SYMVU program for producing three-dimensional graphic displays.

The second alternative involves providing data directly in a matrix form, either on cards or tape.

6.1.2. Required Input

To obtain a graphic display the user must provide three primary control cards and has the option of providing three other sets of control cards and a user subroutine. The required control cards are:

(i) Title Card 1 Card 1 is the title card. One has the flexibility of writing up to 72 characters describing the title of the particular plot that he is generating. It is important that this title be clearly descriptive to differentiate the particular map being requested from all other maps of a similar nature which have been run previously or which may be run in the future;

(ii) Elective Card 2 This card has the primary variables that specify the number of rows and columns of the matrix and the type of view. It also has a number of other variables concerning the input of data, symbols, scales, and the area outside the study area.

Electives 2-1, 2-2, and 2-4 must be specified for the program to run; and,

(iii) Elective Card 3 This card has the primary variables that control the size and viewing angles of a plot. It also has a number of other variables concerning the minimum and maximum value constraints, viewing distances for non-orthogonal views, step sizes, symbol sizes, and block sizes. Electives 3-1, 3-2, 3-3, and 3-4 must be specified for the program to run.

Each card has a different format specification. The first card is completely flexible using the "A" format. The second card requires an "I" format (no decimal point). The third card requires an "F" format (decimal point required) on all electives specified. The content and specifications for each elective are described in the following pages. Examples have been provided to illustrate some of the options within the program.

The optional control cards are used if the data matrix is on cards or if symbols or legends are desired on the plot. Subroutine Data is used to manipulate the values to be plotted or to read in a data matrix on cards.

If elective 21 of the SYMAP program has been used to generate a tape containing the required data matrix, then electives 2-12 and 2-18 must be specified for the program to run.

6.2. ELECTIVE CARD 2

Card 2 has 22 electives. All variables on this card are specified with an "I" format (no decimal point should be used) and should be punched right-justified.

Electives 2-1, 2-2, and 2-4 must be specified for the program to run.

6.2.1. Elective 2-1: Rows (Cols 1-4)

This elective is used to specify the number of rows (excluding border) in the data matrix to be plotted. The maximum number of rows at present is 130. Specify the number of rows in columns 1 through 4 right-adjusted. This elective must be specified or the program will not function correctly.

6.2.2. Elective 2-2: Columns (Cols 5-8)

This elective is used to specify the number of columns (excluding the border) in the data matrix to be plotted. The maximum number of columns at present is 130. Specify the number of columns in columns 5 through 8 right-adjusted. This elective must be specified or the program will not function correctly.

If an incorrect number of rows and/or columns have been entered, one of the following will occur:

(i) If the number punched is less than the correct number on file, only the part of the map specified will be plotted.

(ii) If the number punched is greater than the correct number on file, the program will terminate.

6.2.3. Elective 2-3: View (Col. 12)

This elective is used to specify the type of projection that is desired. The SYMVU program is capable of drawing isometric projections, two-point perspectives, and planometric projections.

The isometric projection is generally the most useful for geographic data. The two-point perspective allows for dramatic visualization, particularly for topographic surfaces on landscape. The planometric view preserves right angles, and therefore the true shape of the map.

To specify isometric projection, punch a one in column 12. If a two-point perspective is desired, punch a two in column 12. If a planometric projection is desired, punch a 3 in column 12. If no specification is made the program assumes that an isometric projection is desired.

6.2.4. Elective 2-4: Line Type (Col. 16)

This elective is used to specify the direction of the lines to be drawn. The program has the capability of plotting the lines along three different axes: along the columns, along the rows, or on either diagonal of the matrix.

Generally, the option of diagonals is specified because this enables the maximum number of lines to be plotted on the drawing. However, for certain azimuth angles, the view looks far better if the lines are drawn along the rows or the columns. Normally any angle chosen between 330 and 30 degrees or 150 and 210 degrees should be drawn along rows whereas, when the viewing angle is between 60 and 120 degrees or between 240 and 300 degrees the lines should be drawn along the columns. All other angles should utilize the diagonal option. Example 6-1 illustrates these optimum azimuth angles for viewing.

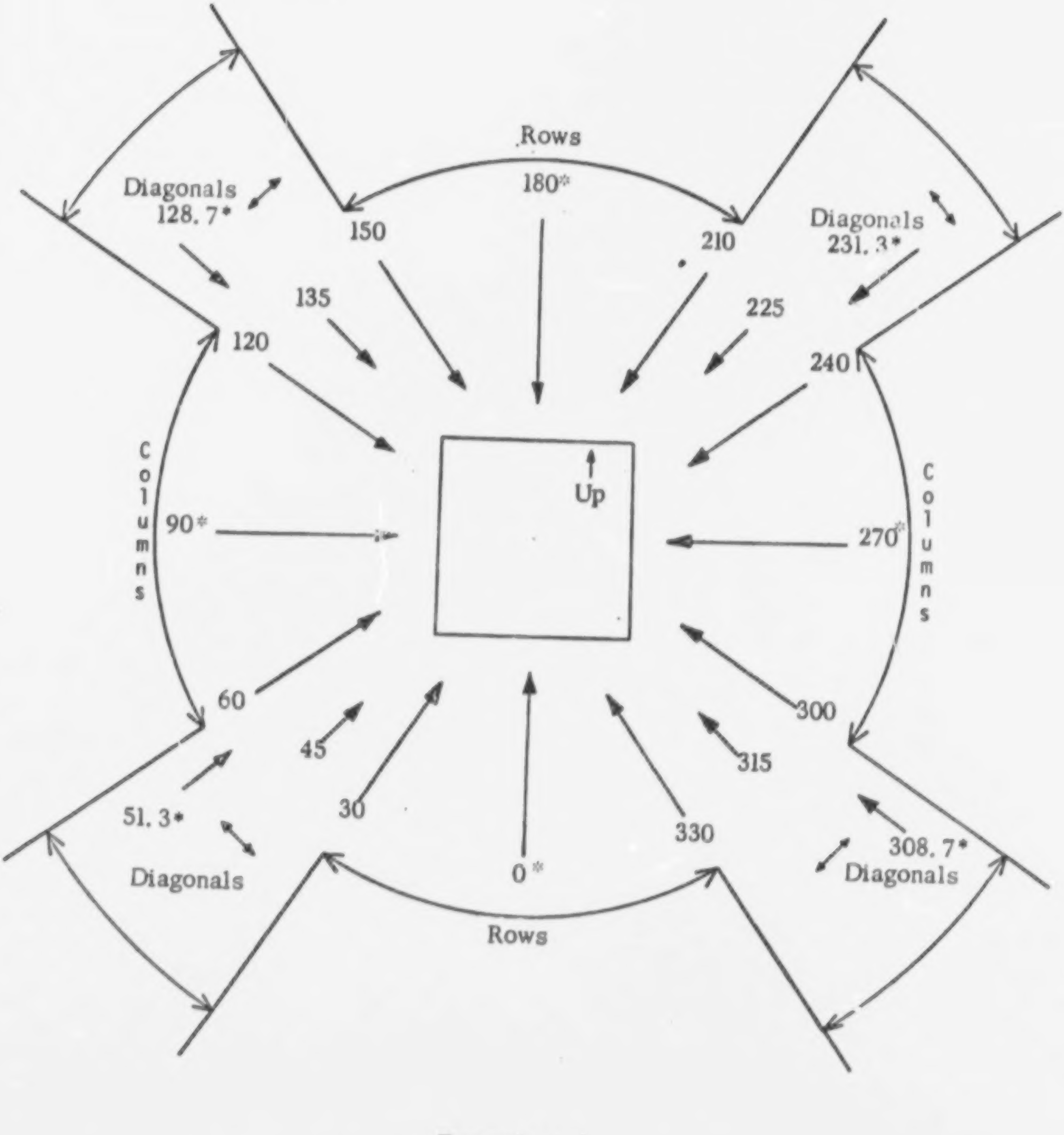
If lines are desired along columns (required if the azimuth number is equal to 90 or 270 degrees) specify one in column 16. If lines are desired along rows (required if azimuth is 0 or 180 degrees) specify two in column 16. If lines are desired along diagonals specify four or five in column 16. This elective must be specified or the program will not function correctly.

6.2.5. Elective 2-5: Smooth (Cols 17-20)

This elective is used to specify the number of binomial smoothings of the data desired. This option is rarely used; it is of some significance for graphically portraying a topographic surface based on discreet point observations which often exhibit great local variance. Specify the number of smoothings desired in columns 17 through 20. Any number of smoothings can be specified, but eventually the surface will look like a flat plane.

6.2.6. Elective 2-6: Repeat (Col. 24)

This elective is used to specify whether the data from the previous plot is to be retained, or a new data set read in. The program can use the same data for a number of plots with


Rows
180*
Diagonals
128.7*
150
210
Diagonals
231.3*
135
225
120
240
Columns
90*
Up
270*
Columns
60
300
45
315
30
330
51.3*
308.7*
Diagonals
0*
Diagonals
Rows

Example 6-1
Azimuth Orientation System

Example 6-1
(Continued)

* A view chosen from any of these angles will produce plotted lines which are perpendicular to the lines of sight and are therefore parallel to the bottom edge of the paper. The appropriate line type must be specified:

Angle	Line Type	
0, 180	rows	2
90, 270	columns	1
51.3, 231.3	diagonals	5
128.7, 308.7	diagonals	4

Note also that the diagonal type to be used for angles between 30 to 60 and 210 to 240 is "5", whereas the type between 120 to 150 and 300 to 330 is "4 ". When angles such as 51.3 and 128.7 are specified in Elective 3-2: Azimuth, the angle as printed on the output will be rounded off to the nearest whole number: 51 and 129.

different angles and elevations. To read in a new data set, specify a zero in column 24; to use a previously read data set, specify a one in column 24. If no specification is made the program assumes that the plot is to be drawn utilizing a new data set.

6.2.7. Elective 2-7: Lnint (Cols 25-28)

This elective is used to specify which of the lines are to be plotted. Specify two for every second line, three for every third, etc. as an integer right-justified in columns 25-28. If no specification is made the program assumes every line should be plotted.

6.2.8. Elective 2-8: Nozero (Col. 32)

This elective is used to specify the symbolism outside of an irregular outline study area. When the SYMAP program is used for generating SYMVU data, an irregular outline often occurs instead of a rectangular grid of matrix values.

This elective allows for flexibility in determining what should be done with the areas outside of the outline. One may choose to plot the non-study area with lines; in this case the area appears as a flat surface. A second option is to delineate the non-study area with both flat lines and a shading of symbolism such as "x's"; this usually gives the appearance of a very solid or heavy base. A third alternative is to delete the non-study area; this is visually the weakest in that the illustration appears to be a suspended surface. (Note that when the non-study area is plotted with lines, the program assigns zero values to the part of the matrix outside the study area. This results in the "up lifted" surface appearance on study areas having irregular outlines.)

To plot the non-study area as 'zero lines', specify zero in column 32. To delete the non-study area, specify 1 in column 32. To darken the non-study area with light symbolism so as to look white specify 2 in column 32; to darken the non-study area with dark symbolism, specify 3 in column 32. If no specification is made the program assumes that a non-study area should be plotted as zero lines.

6.2.9. Elective 2-9: Sqrtt (Col. 36)

This elective is used to specify that the square roots of all values in the data matrix should be calculated and used as the values to be plotted. This has the effect of reducing the relative importance of extreme values. This particular elective is useful for a large range of data where smoothing is possible not only for the higher values but also for the lower values. (For example, if one is studying the population density in a region surrounding New York City, the peaks of population on Manhattan Island would have the effect of blocking out any detail occurring in the areas surrounding the city. By taking the square root of the data the surrounding surface variations would appear more distinguishable.)
To take the square root of all values, specify a 1 in column 36.

6.2.10. Elective 2-10. Sameh (Col. 40)

This elective is used to specify the same vertical scale factor as the preceding plot. It is often employed when a number of plots are being made in a time series of data and it is desirable to have maps comparable in vertical scale. To specify that the same "Z" scale factor should be used as was used on the preceding plot, place a 1 in column 40.

6.2.11. Elective 2-11: Endlin (Col. 44)

This elective is used to specify that the endlines circumscribing the plot be deleted. To delete the endlines, specify 1 in column 44.

6.2.12. Elective 2-12: Symap (Cols. 45-48)

This elective is used to specify the number of vertical lines per inch contained in the input matrix.

If the data matrix was produced by the SYMAP program, the output is normally in an 8 by 10 grid module; there are eight vertical lines per inch. The module of the SYMAP program is based on the two dimensions of the character symbols common in most line printer devices. SYMVU has the capability of reading in data which has a vertical dimension of 6 lines, 8 lines, or 10 lines per inch. This flexibility allows for types of input other than the SYMAP program. If the GRID program is used, an equal-sized cell is the module of the matrix. In this case equal scales for the horizontal and vertical dimensions of the matrix should be specified for SYMVU.

To specify the number of lines per inch, punch either a 6, 8, or 10 right-adjusted in columns 45 through 48. If no specification is made the program assumes equal scales for the horizontal and vertical dimensions of input.

6.2.13. Elective 2-13: Leg (Col. 50)

This elective is used to specify the deletion of the control information normally written out on a SYMVU plot. This information includes the elevation angle, the azimuth angle, and the scaling information. To delete this information from the plot, specify a 1 in column 50.

6.2.14. Elective 2-14: Nscal (Col. 52)

This elective is used to specify that a height scale legend should appear to the right of the plot. This scale has measurements in inches written on the left side and the corresponding "Z" value on the right side. At the top of the scale the greatest height and the corresponding "Z" value are shown. To call for this height scale legend, specify a 1 in column 52.

6.2.15. Elective 2-15: Flp (Col. 56)

This elective is used to specify that a floating plane is desired as part of the plot. To include a floating plane, specify a 1 in column 56.

6.2.16. Elective 2-16: Base (Col. 60)

This elective is used to specify whether a base on which a topographic surface can rest should be deleted or constructed. When the data for this type of surface is in a rectangular grid matrix (no irregular outlines) the bottom line of the base is determined by the lowest value plotted in the surface. When an outline is used within the matrix (zero values are being plotted), the program determines the appropriate vertical dimension of the base. This base can help in identifying the geographic relationships of the study area.

To delete the base from the plot, specify a 1 in column 60. To include a complete base underneath an irregular outline, specify a 2 in column 60. To place the surface on a block, specify a 3 in column 60. Use Elective 3-10: Size to specify the height of this block. If no specification is made the program assumes an irregular base under an irregular outline; when a full rectangular matrix of values is being plotted, the program will normally construct a base on which the total surface can rest. In this case the base is determined by the lowest value being plotted.

6.2.17. Elective 2-17: Pltdot (Cols. 61-64)

This elective is used to specify the number of symbols that will be located on the plot. The SYMVU program has the capability of plotting certain symbols geographically on the surface; normally the program is limited in vocabulary to five types of symbols. (The program can be altered slightly to change or manipulate the size and type of symbol as discussed in Elective 3-9).

Symbols are often used for locating certain phenomena such as political boundaries, population concentrations or other geographic features. They have also been used for drawing legends such as north arrows. This elective specifies the number of symbols or plot dots that the program is to read in. The exact coordinate or geographic locations by row and column, as well as the actual symbol type for each symbol, are specified on a series of data input cards following the three main control cards. These are discussed in the section on optional data cards. The symbol types are shown in Table 6-1.

Specify the number of symbol points to be used, right-adjusted in columns 61 through 64. If no specification is made the program assumes that there are no symbol points to be plotted.

6.2.18. Elective 2-18: Fdata (Cols 65-68)

This elective is used to specify the data input type and file number. The SYMVU program uses data from SYMAP output unaltered and other data through a user-written subroutine.

Normally a tape is created by Elective 21 of the SYMAP program which contains the data matrix to be plotted. A number of data matrices are often created on the same tape with each matrix specified on a separate file. Elective 2-18 of the SYMVU program is used to specify the file on the tape that is to be plotted.

If the data matrix is to be supplied through a user-written subroutine, specify a zero right-adjusted in columns 65 through 68. If the data matrix is on a SYMAP output unit, specify the file number right-adjusted in columns 65 through 68. If no specification is made the program assumes that the data matrix will be read in by a user subroutine.

6.2.19. Elective 2-19: Namin (Col. 72)

This elective is used to specify that the values falling below a predetermined minimum should be set equal to that minimum value. Elective 3-5 is used to set the minimum value to be used by the program.

Specify a 1 in column 72 to set the values below a predetermined minimum value to that minimum. Elective 3-5 will be ignored if it is used without this elective. If no specification is made the program assumes the actual values should be plotted.

6.2.20. Elective 2-20: Namax (Col. 76)

This elective is used to specify that the values falling above a predetermined maximum should be set equal to that maximum value. Elective 3-6 is used to set the maximum value to be used by the program.

Specify a 1 in column 76 to set the values above a predetermined maximum value to that value. Specify a 2 in column 76 if the values above the maximum specification are not to be changed but the maximum value specified in Elective 3-6 is to be used in conjunction with the height in Elective 3-4. Elective 3-6 will be ignored if it is used without this elective. If no specification is made the program assumes the actual values should be plotted.

When the value extremes are so great as to minimize the variation in the rest of the data values both electives 2-19 and 2-20 should be used.

6.2.21. Elective 2-21: Iform (Cols. 77-78)

This elective is used to specify a value which may be used in SUBROUTINE DATA to identify the appropriate data set to be manipulated. Specify an integer number (from 1 to 99) in columns 77-78. If no specification is made the program assumes that no more than one data set is to be manipulated in SUBROUTINE DATA.

6.2.22. Elective 2-22: Nleg (Cols. 79-80)

This elective is used to specify the number of text legends to be plotted in each SYMVU. The maximum number is 50. The text legends are discussed in the section on optional data cards.

Specify the number of text legends to be plotted right-adjusted in columns 79-80. If no specification is made the program assumes that there are no text legends to be plotted.

6.3. ELECTIVE CARD 3

Card 3 has 10 electives. All variables on this card are specified with an "F" type format (they require a decimal point in all field specifications). They should be punched right-justified; this is not mandatory as it is with the "I" format on card 2.

Electives 3-1, 3-2, 3-3, and 3-4 must be specified for the program to run.

6.3.1. Elective 3-1: Altitude (Cols. 1-5)

This elective is used to specify the elevation of the viewing position above the horizontal plane. Specify the viewing position in angle degrees in columns 1 through 5. This elective must be specified or the program will not function correctly.

For general purposes, a good height to begin experimentation with is 45 degrees. For most cases, this gives an optimum view of the surface as well as a good relationship between the lines and the white space which separate them. A zero to five degree view of a landscape topographic surface normally is quite close to eye level viewing whereas, those views nearer to 45 degrees appear to the observer as birds-eye views.

For most purposes visual variation in the topographic surface is primarily a function of the white space between the delineated lines. Because the only variation in line direction occurs in the vertical dimension the maximum variation in a line occurs at the altitude of zero degrees; generally, this angle hides much of the study area because of the hidden line factor in the program.

6.3.2. Elective 3-2: Azimuth (Cols. 6-10)

This elective is used to specify the desired horizontal angle of direction for viewing. Specify the azimuth viewing angle in degrees right-adjusted in columns 6 through 10. This elective must be specified or the program will not function correctly.

The SYMVU program can rotate any data matrix 360 degrees when using an azimuth orientation system. When one specifies a zero azimuth, he is looking perpendicular to the bottom of the study area. As he rotates his viewing position towards the left, he is increasing toward 360 degrees. At 90 degrees, he is looking perpendicular to the left hand side of the matrix. At 270 degrees, he is looking at the right hand side of the matrix, etc. Example 6-1 shows the azimuth orientation system used for SYMVU.

6.3.3. Elective 3-3: Width (Cols. 11-15)

This elective is used to specify the width, in inches, that the input matrix will have on the plot. There are two scaling variables in the SYMVU program: the width and height. The width refers to the distance along the bottom side of the matrix (the columns of the input matrix). Under normal situations, the width dimension and the height dimension when combined, should not exceed 11 inches.

Specify the width in inches right-adjusted in columns 11 through 15. This elective must be specified or the program will not function correctly.

6.3.4. Elective 3-4: Height (Cols. 16-25)

This elective is used to specify the height, in inches, that the input matrix will have on the plot. Specify the height in inches right-adjusted in columns 16 through 25. If you specify 0.0, the height will be set to 3.0 inches. Any practical height greater than 0.0 may be specified, but the combination of width and height should not exceed 11 inches. This elective must be specified or the program will not function correctly.

6.3.5. Elective 3-5: Amin (Cols. 26-35)

This elective is used to specify the minimum value that the program should use for plotting. The user normally specifies this elective when an extreme minimum value point occurs within the data matrix. Specify the minimum value desired right-adjusted in columns 26 through 35. If no specification is made the program assumes that the actual minimum value should be used for plotting. This elective is ignored unless Elective 2-19 is also used.

6.3.6. Elective 3-6: Amax (Cols. 36-45)

This elective is used to specify the maximum value that the program should use for plotting. When extremely high values occur within the data matrix, this elective is used for controlling or truncating those values. Specify the maximum value desired, right-adjusted in columns 36 through 45. If no specification is made the program assumes that the actual maximum value should be used for plotting. This elective is ignored unless Elective 2-20 is also used.

In connection with Elective 2-20 and Elective 3-4, this elective may be used to set the height equal to some specified value (the maximum value) but allow the actual maximum value to be used for plotting. This is particularly useful for a set of plots where a uniform value scale is desired.

6.3.7. Elective 3-7: Vdist (Cols. 46-50)

This elective is used to specify the viewing distance for perspective drawings. The option controls the distance from which one observes a given perspective. As one moves back from the illustration (in other words, increases his viewing distance) the ends or the corners of the drawing are usually spread out farther relative to the center line. With a shorter viewing distance the focus is primarily on the center of the data matrix. This variation is most readily seen when the altitude angle is very low, such as when the altitude is near eye level on a natural landscape drawing. When the altitude is very high, such as 45 or 50 degrees, variation caused by the viewing distance is relatively small.

Specify the viewing distance in inches in columns 46 through 50 right-adjusted. If no specification is made the program assumes a viewing distance of 10.0 inches; this option is used only with perspective views.

6.3.8. Elective 3-8: Step (Cols. 51-55)

This elective is used to specify non-standard plotter step size. This elective is rarely used but can be employed to reduce calculating time with a commensurate loss in detail and smoothness of line segments. Specify the step size in inches in columns 51 through 55 right-adjusted. If no specification is made the program assumes a standard step size of 0.005 inches.

6.3.9. Elective 3-9: Sepsm (Cols. 56-60)

Elective 3-9 This elective is used to specify non-standard sizes of the symbols (plot dots or legends) used on the drawing. Specify the actual size in inches right-adjusted in columns 56 through 60. For normal purposes, the standard scale factor is adequate. With the exception of symbol number 14 which is larger, (see Table 6-1) the standard symbol size is 0.25 inches. Occasionally for specialized purposes, the user may find it advantageous to vary the size of the symbol he is working with. If no specification is made the program assumes a standard symbol size.

6.3.10. Elective 3-10: Size (Cols. 76-80)

This elective is used to specify the height of a block placed under the study area. This elective is used along with elective 2-16 (BASE) when BASE=3. Specify right-justified in columns 76-80 the height you want the block in inches up to a maximum height of 0.75 inches. If no specification is made the program assumes a standard height of 0.5 inches when BASE=3 in Elective 2-16.

6.4. OPTIONAL DATA CARDS

There are a number of optional data cards that can be specified for the SYMVU program:

(i) If the data matrix is to be read in from cards instead of tape, for each row of the matrix one or more cards are needed containing the values for the columns in the row;

(ii) For each symbol to be plotted a card is needed containing the location and symbol type; and,

(iii) For each legend to be plotted a card is needed containing the location and legend text.

These cards follow directly after Elective Card 3 and must be in the above order.

6.4.1. Data Matrix on Cards

If the data matrix is to be submitted to the computer from cards instead of tape, the card deck for the entire matrix follows immediately after the third control card. Normally, the program uses tape inputs.

The cards are read in under a format specified in Subroutine Data. There will be one record--on one or more cards--for each row sequentially from the top to the bottom of the matrix; each record will contain a value for each column sequentially from the left to the right of the matrix.

6.4.2. Symbol Points

For each symbol to be plotted, the row measurement, the column measurement and the symbol 'type' must be specified on one card. Coordinate measurements must be in terms of the input grid to SYMVU. If SYMAP is used to create the input tape to SYMVU, these measurements in rows and columns should be copied from the table immediately preceding the map produced by SYMAP, to plot data points.

Specify the row measurement right-adjusted in columns 1 through 10 using an "I" format (no decimal point) and the column measurement in columns 11 through 20, also right-adjusted using an "I" format. The symbol type should be specified in columns 29 through 30, right-adjusted using an "I" format. Table 6-1 shows the symbols that may be used. If columns 29 through 30 are left blank, the default symbol is 14.

The total number of plot dots must be specified in Elective 2-17.

6.4.3. Legends

For each legend to be plotted, the row measurement, the column measurement, and the actual legend text must be specified on one card. Coordinate measurements must be in terms of the input grid to SYMVU. If SYMAP is used to create the input tape to SYMVU, these measurements in rows and columns should be taken directly from the map produced by SYMAP.

Specify the row measurement right-adjusted in columns 1 through 10 using an "I" format (no decimal point) and the column measurement in columns 11 through 20, also right-adjusted using an "I" format. Columns 30 through 54 are

Table 6-1
SYMBOL POINTS

If you want this symbol	Specify in Columns 29-30
△	02
+	03
✳	11
☆	14
%	21

If columns 29-30 are left blank the default symbol is 14.

used for the desired text legend; however, this should be left-justified so that the first legend character will be in the column specified in columns 11 through 20.

The total number of legends must be specified in Elective 2-22.

6.5. SUBROUTINE DATA

This Fortran Subroutine can be used to manipulate the Z-values and return the data to the program in the form that SYMVU will accept. Subroutine Data is also used to read in data from a data matrix on cards.

```
Columns   1111111111222222222233333333334
1234567890123456789012345678901234567890

      SUBROUTINE DATA(X,IR,IC,IFORM)
      DIMENSION X(130)
      READ(5,1000) (X(I),I=1,130)
1000  FORMAT(40F2.0/40F2.0/40F2.0/10F2.0)
      DO 10I=1,130
  10  X(I)=X(I)*2-1.
      RETURN
      END
```

For the example shown:

-- X(130) returns the values for one row of the matrix at a time to the program;

--IR is the number of the row to be read in and returned;

--IC is the number of columns (or Z-values) per row, read in by Elective 2-2; and,

--IFORM is the user supplied value which identifies the data set to be manipulated by a particular set of statements in Subroutine Data. (In this case, only one data set is manipulated and IFORM is not used) IFORM is read in by Elective 2-21.

The values for each row (X(I), I=1, 130) are read in from a data matrix on cards (read from unit 5, the card reader) according to the format in statement 1000 (four cards per row). The values of x are manipulated (X(I)=X(I)*2-1.) before being returned to the program.

6.6. SAMPLE SUBMISSIONS

Examples 6-2 through 6-7 show six sample SYMVUs. Table 6-2 summarizes the electives used for these SYMVUs. The first three show three-dimensional views for the types of maps that can be produced by SYMAP: conformant, contour, and proximal. The last three show the types of views possible with SYMVU: isometric, perspective, and planometric.

These SYMVUs are not intended to demonstrate the best choices or combinations of electives in the program. It is the intention, however, to show as many different electives as possible so that the user will be provided with a good basis upon which the wisest judgements can be made. Many of the SYMVU electives have been kept constant in order that the non-required and lesser-used electives may be best displayed. Finally, Example 6-8 summarizes all the program electives and their uses. This listing is included with the program.

There are a few things to keep in mind when producing more than one SYMVU in a single submission:

(i) Any view of the same surface after the first view can be requested merely by including the three control cards immediately after the control cards (and after any optional data cards) for the previous view;

(ii) When requesting a view of a new surface--whether it be the first, second, etc., surface in the submission--all optional control cards must be re-specified with the exception of the data matrix. In other words, symbol point and legends are not carried forward from one surface to the next, although they are carried forward from one view of a surface to the next view of the same surface;

(iii) In addition to the required electives, two other electives generally change from view to view and surface to surface and should be reviewed carefully: Elective 2-6: Repeat and Elective 2-18: Fdata. Elective 2-12: SYMAP and Elective 2-21: IFORM specify whether the input is on cards, tape, or disk--usually on a file produced by SYMAP. Finally Elective 2-17 (Pltdot) and Elective 2-22 (Nleg) are used when symbol points and legends are involved;

(iv) These principles are summarized for a hypothetical submission where;

-- the first surface is on disk, two views are desired, and there are no symbol points or legends,

-- the second surface is on cards (Subroutine Data would be used to read the data in), two views are desired, and there are legends but no symbol points;

-- the third surface is also on disk (it must be on a file that occurs after the file containing the first surface--rewinding in the middle of a run is not possible here), three views are desired, the same legends are to be used and symbol points are added.

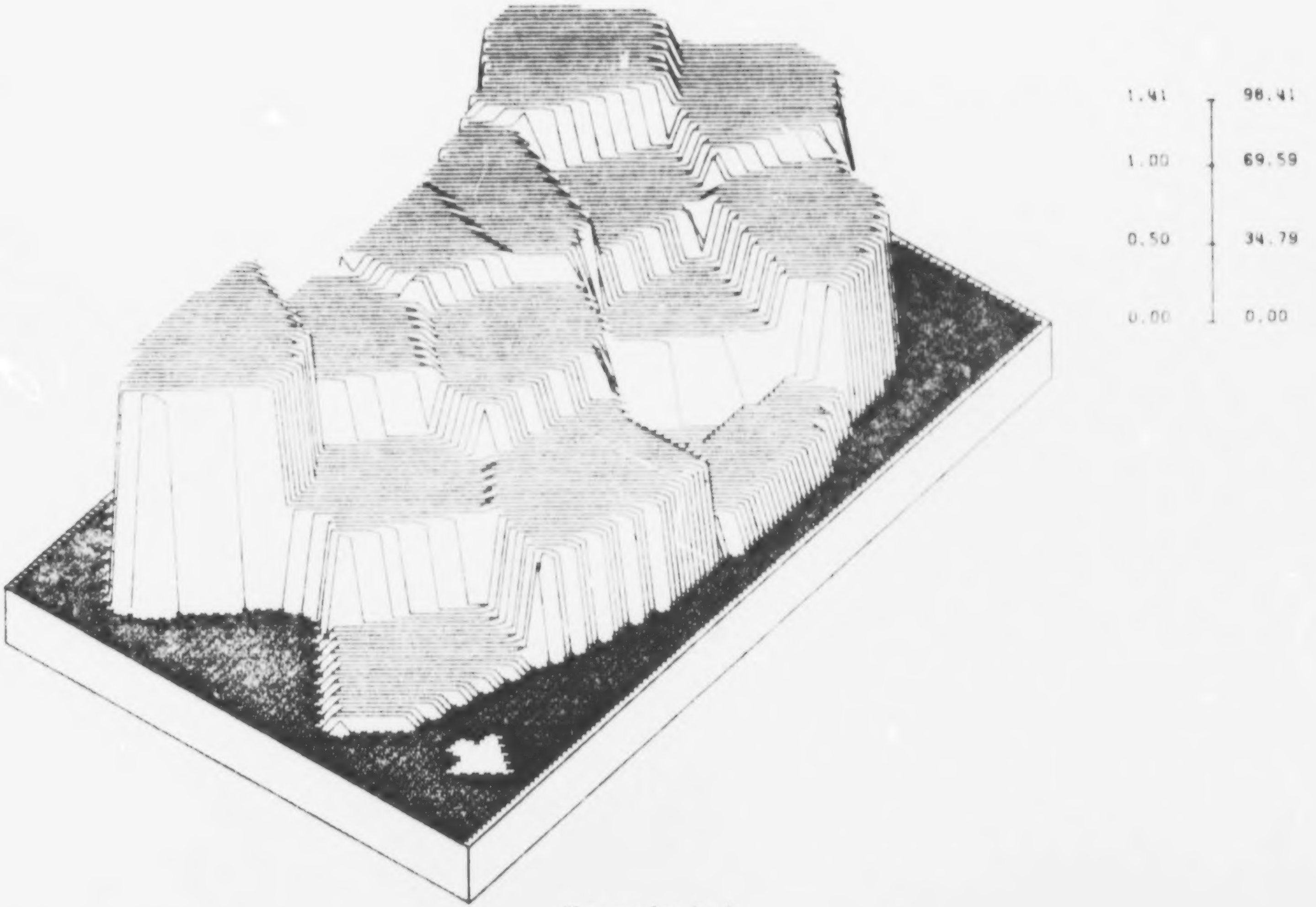
```
               Control Cards     .
               Subroutine Data   .
               Control Cards     .
               Data              .  TITLE CARD 1          . first surface, first view
                                    ELECTIVE CARD 2
                                    ELECTIVE CARD 3
Data input for first sur-                                 . first surface, second
face on tape or disk             .  TITLE CARD 1            view
No symbols or legends            .  ELECTIVE CARD 2
                                    ELECTIVE CARD 3

                                    TITLE CARD 1          . second surface, first
                                    ELECTIVE CARD 2         view
                                    ELECTIVE CARD 3
Data input for second
surface on cards                 .    - - - -
No symbol points                 .    - - - -
Legend cards                     .    - - .......
                                      - - .......
                                    TITLE CARD 1          . second surface, second
                                    ELECTIVE CARD 2         view
                                    ELECTIVE CARD 3
                                    TITLE CARD 1          . third surface, first
                                    ELECTIVE CARD 2         view
Data input for third                ELECTIVE CARD 3
surface on tape or disk          .    - - .
Symbol point cards               .    - - .
                                      - - .
Legend cards                     .    - - .......
 (same as second                      - - .......
    surface)

                                    TITLE CARD 1          . third surface, second
                                    ELECTIVE CARD 2         view
                                    ELECTIVE CARD 3
                                    TITLE CARD 1          . third surface, third
                                    ELECTIVE CARD 2         view
                                    ELECTIVE CARD 3
```

TABLE 6-2
ELECTIVES USED IN EXAMPLES

ELECTIVE	(name)	-EXAMPLE- 6-2	6-3	6-4	6-5	6-6	6-7
2-1	ROWS	63	63	63	63	63	63
2-2	COLUMNS	124	124	124	124	124	124
2-3	VIEW	0*	0*	0*	1	2	3
2-4	TYPE	4	4	4	2	2	2
2-5	SMOOTH						
2-6	REPEAT						
2-7	LNINT						
2-8	NOZERO	2	2	2	1		2
2-9	SQRTT						
2-10	SAMEH						
2-11	ENDLIN						1
2-12	SYMAP	8	8	8	8	8	8
2-13	LEG						
2-14	NSCAL	1	1	1	1	1	1
2-15	FLP					1	
2-16	BASE	3	3	3		3	
2-17	PLTDOT	31	46	46		31	15
2-18	FDATA	1	2	3	2	2	2
2-19	NAMIN				1		
2-20	NAMAX				2		
2-21	IFORM						
2-22	NLEG				3		
3-1	ALTITUDE	45.	45.	45.	45.	45.	45.
3-2	AZIMUTH	231.3	231.3	231.3	108.	340.	0.0
3-3	WIDTH	6.	6.	6.	6.	6.	6.
3-4	HEIGHT	2.	2.	2.	2.	2.	2.
3-5	AMIN				10.		
3-6	AMAX				75.		
3-7	VDIST					12.	
3-8	STEP						
3-9	SEPSM	.03	.04	.04	.04	.05	.05
3-10	SIZE						

Note: If elective is blank for an example, this elective is not used.

* This elective could have been left blank.

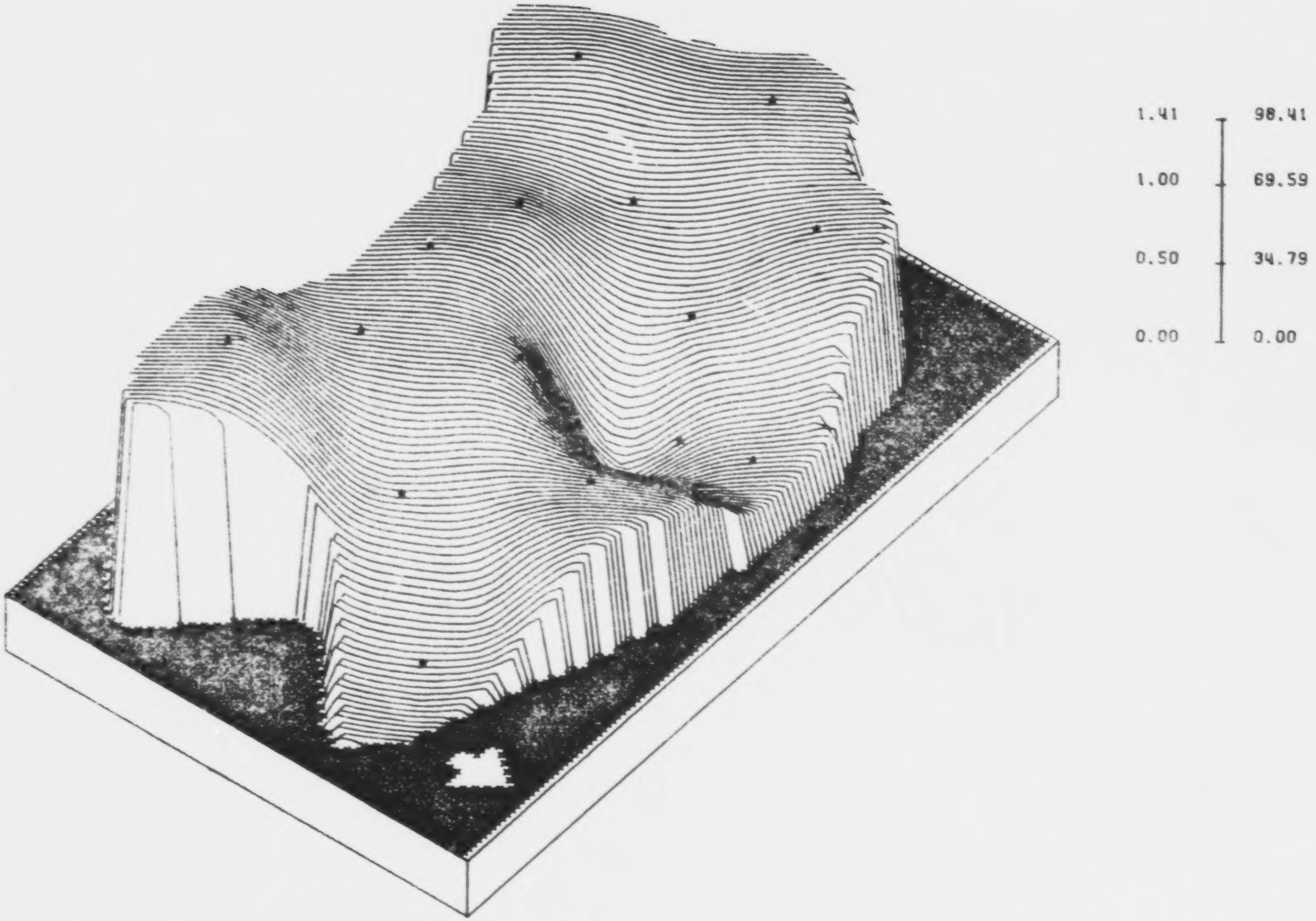


Example 6-2

FIGURE A: MANTEGNA BAY - CONFORMANT

AZIMUTH = 231 ALTITUDE = 45
*WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 09/25/70

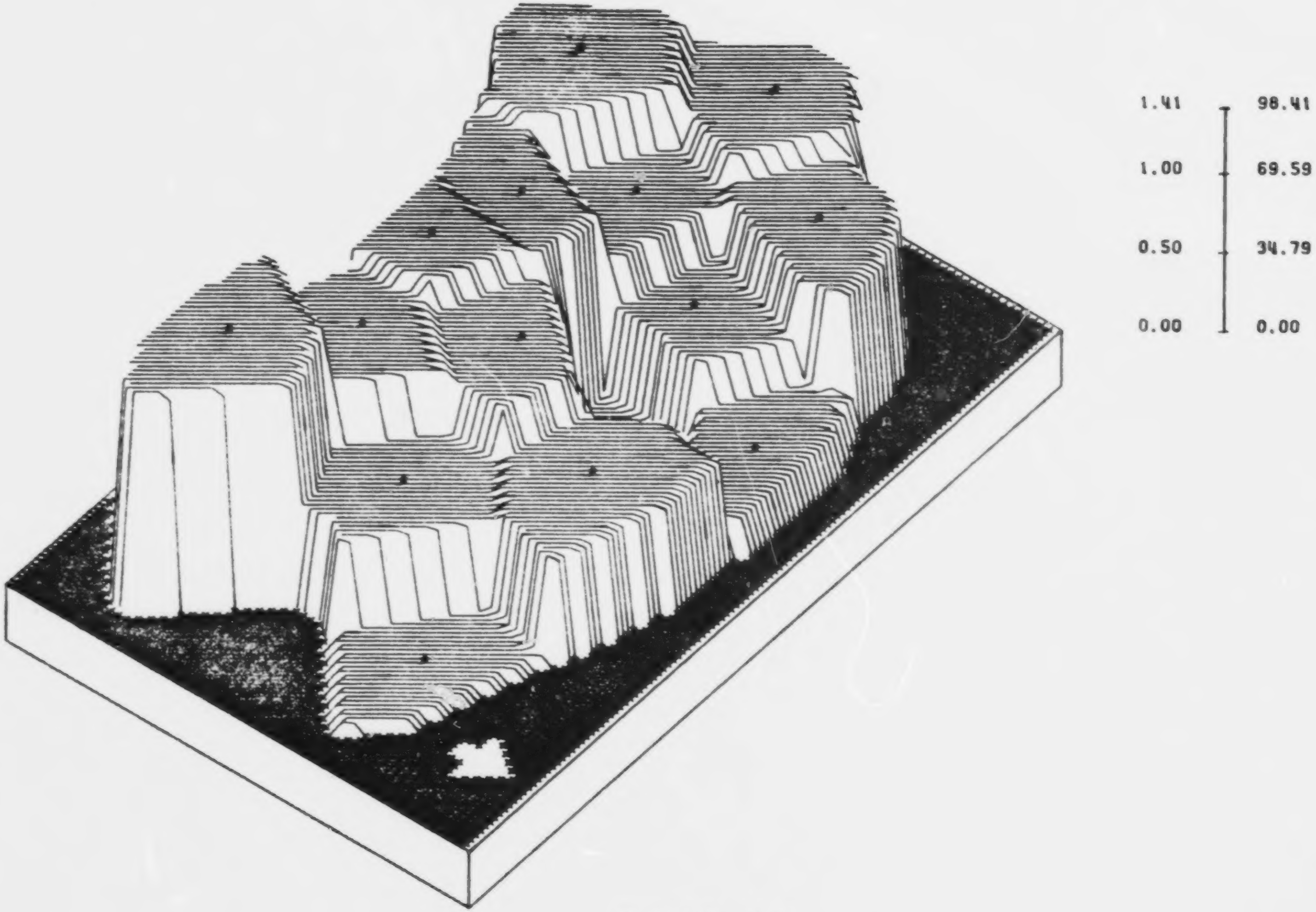

1.41
98.41
1.00
69.59
0.50
34.79
0.00
0.00

Example 6-3

FIGURE B: MANTEGNA BAY - CONTOUR

AZIMUTH = 231 ALTITUDE = 45
*WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 09/25/70

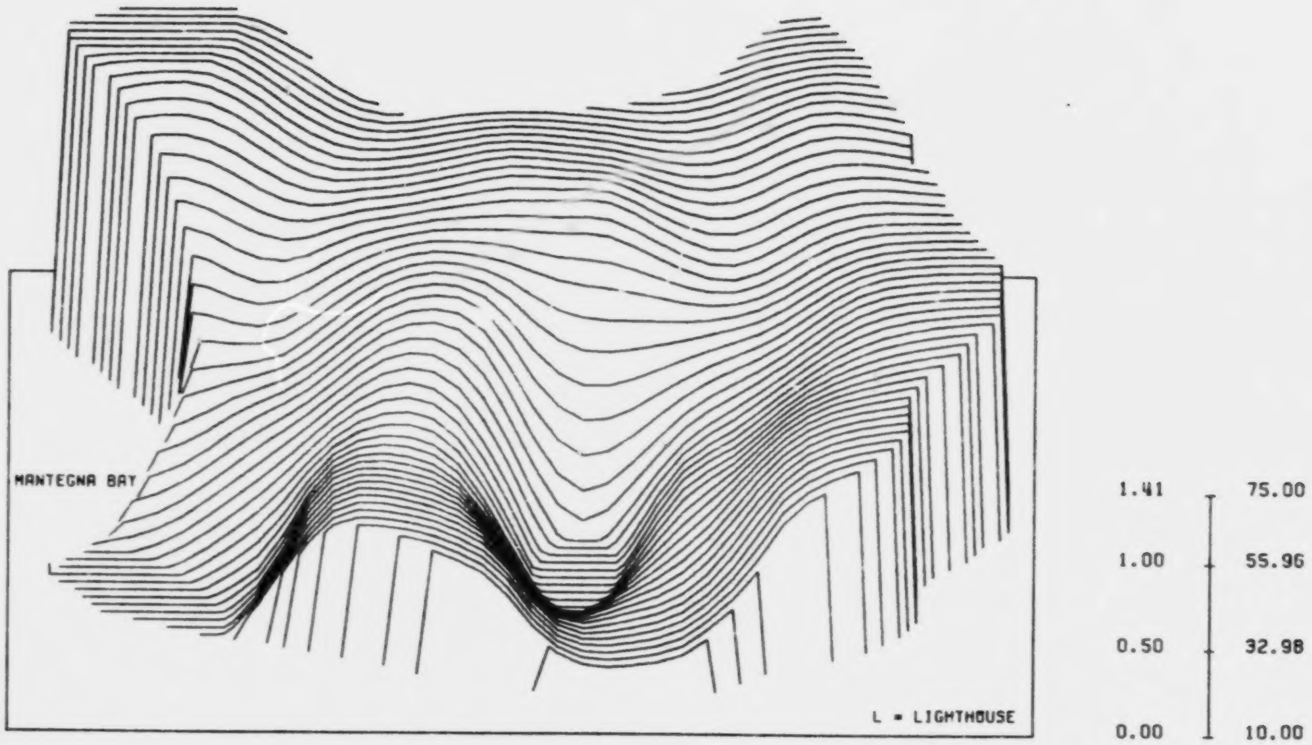

1.41
98.41
1.00
69.59
0.50
34.79
0.00
0.00

Example 6-4

FIGURE C: MANTEGNA BAY - PROXIMAL

AZIMUTH = 231
*WIDTH = 6.00
ALTITUDE = 45
*HEIGHT = 2.00

* BEFORE FORESHORTENING 09/25/70

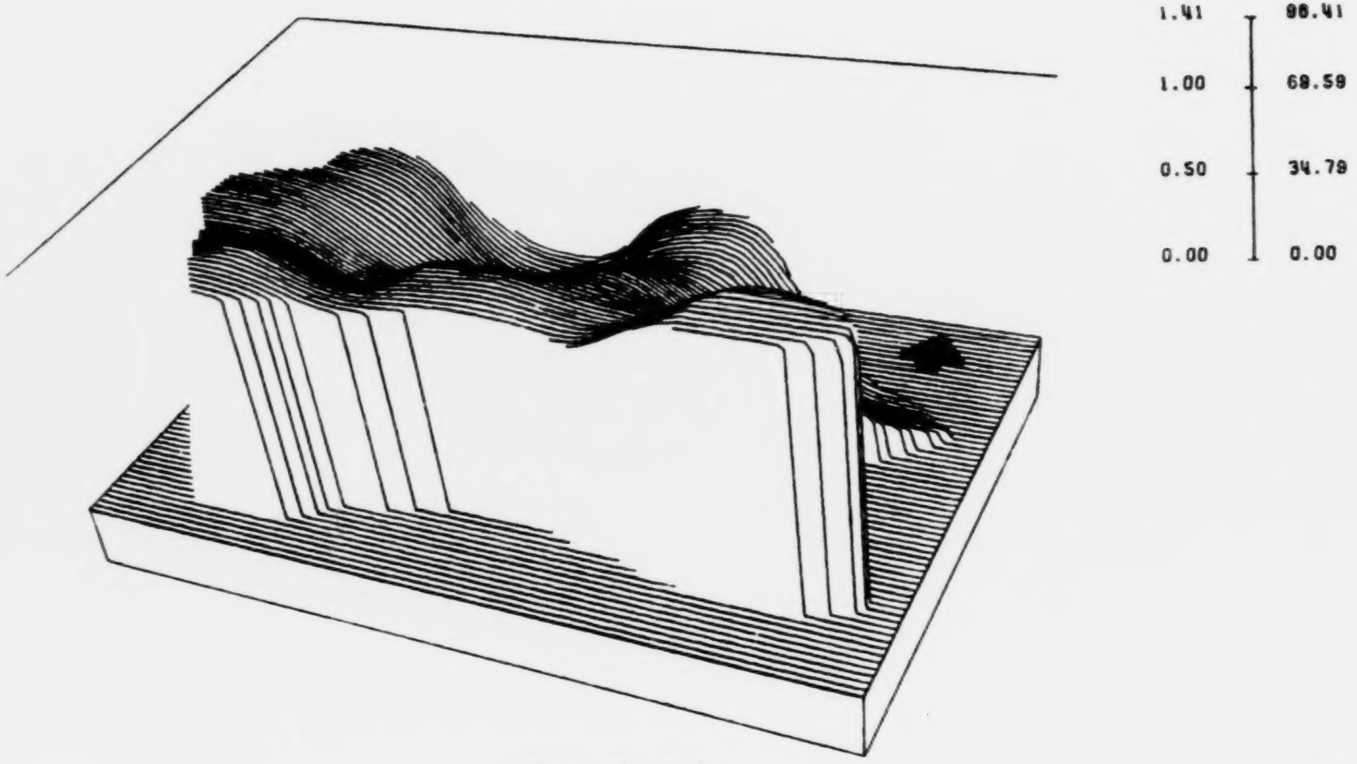


Example 6-5

FIGURE 2: MANTEGNA BAY - ISOMETRIC PROJECTION

AZIMUTH = 180 ALTITUDE = 45
*WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 10/26/70


1.41
1.00
0.50
0.00
98.41
68.59
34.78
0.00

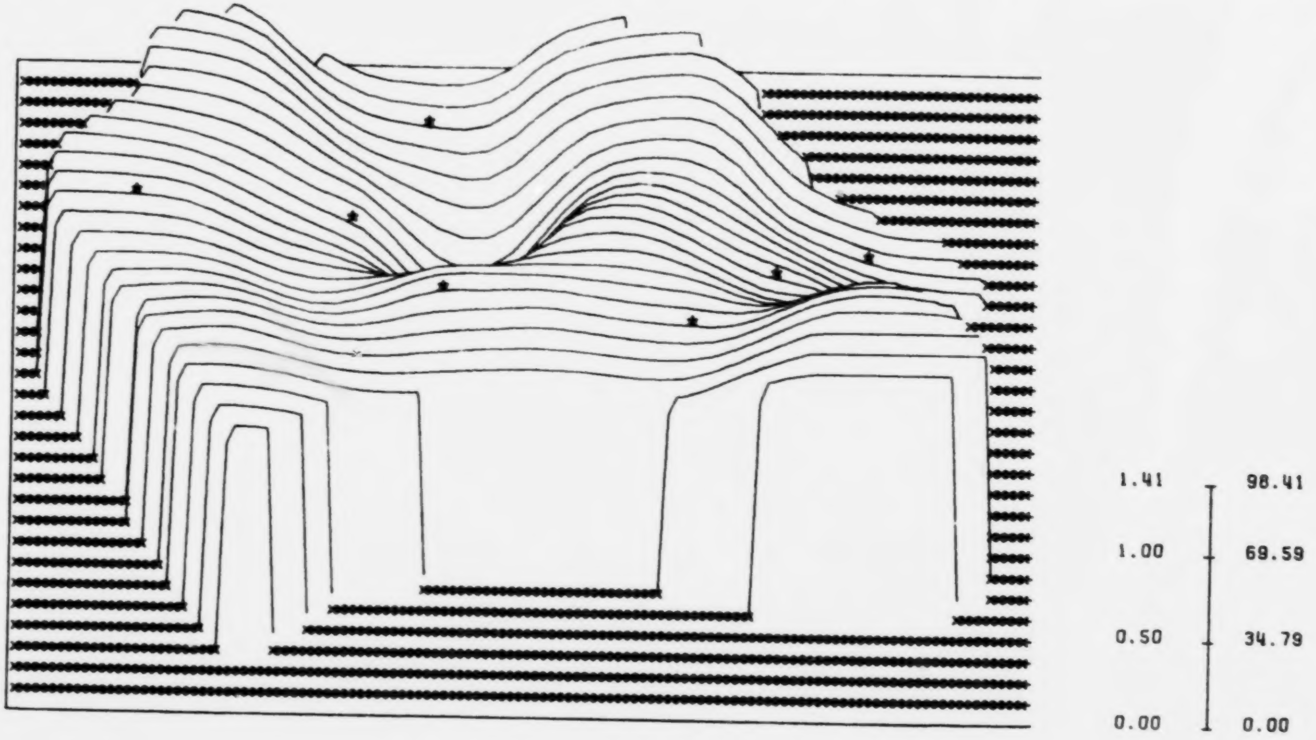
Example 6-6

FIGURE 3: MANTEGNA BAY - TWO POINT PERSPECTIVE

AZIMUTH = 340
*WIDTH = 6.00

ALTITUDE = 45
*HEIGHT = 2.00

* BEFORE FORESHORTENING 10/07/70



Example 6-7

FIGURE 4: MANTEGNA BAY - PLANOMETRIC PROJECTION

AZIMUTH = 0 ALTITUDE = 45
WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 10/06/70

```
           C ************************************************************************ 00000100
           C *                              SYMVU                                   * 00000110
           C * ..........OBLIQUE VIEW PLOTTING PROGRAM FROM GRIDDED DATA.........   * 00000200
           C *                                                                      * 00000300
           C * WRITTEN BY FRANK J. RENS                                             * 00000400
           C *             LABORATORY FOR COMPUTER GRAPHICS AND SPATIAL ANALYSIS    * 00000500
           C *             HARVARD UNIVERSITY,CAMBRIDGE, MASSACHUSETTS              * 00000600
           C *                                                                      * 00000700
           C *                              VERSION 1.0                             * 00000750
           C *                              NOVEMBER, 1970                          * 00000755
           C *                                                                      * 00000800
           C *                                                                      * 00000900
           C * .......PROGRAM INSTRUCTIONS............................................* 00001000
           C *                                                                      * 00001100
           C * REQUIRED DATA CARDS                                                  * 00001200
           C * .....................                                                * 00001300
           C * CARD 1 - TITLE IN COLUMNS 1-72                                       * 00001400
           C *                                                                      * 00001500
           C * CARD 2 - CONTROL CARD (1 OF 2)  -  INTEGER                           * 00001600
Elective:  C *    VARIABLE  COLUMNS    TYPE                DESCRIPTION              * 00001700
  2-1      C *      ROWS      1-4        I          NUMBER OF ROWS IN DATA MATRIX   * 00001800
           C *                                      MAXIMUM OF 130                  * 00001900
  2-2      C *      COLS      5-8        I          NUMBER OF COLUMNS IN DATA       * 00002000
           C *                                      MATRIX, MAXIMUM OF 130          * 00002100
  2-3      C *      VIEW      12         I          0 OR 1 = ISOMETRIC              * 00002200
           C *                                      2 = TWO POINT PERSPECTIVE       * 00002300
           C *                                      3 = PLANOMETRIC                 * 00002400
  2-4      C *      TYPE      16         I          1 = LINES ALONG COLUMNS         * 00002500
           C *                                        REQUIRED IF AZ=90 OR 270      * 00002600
           C *                                      2 = LINES ALONG ROWS            * 00002700
           C *                                        REQUIRED IF AZ=0 OR 180       * 00002800
           C *                                      3 = HISTOGRAMS                  * 00002900
           C *                                       (NOT AVAILABLE)                * 00003000
           C *                                      4 = DIAGONALS VIEWED            * 00003100
           C *                                       FROM (0.,0.)                   * 00003200
           C *                                      5 = DIAGONALS VIEWED FROM       * 00003300
           C *                                       (XMAX,0.) OR (0.,YMAX)         * 00003400
           C *                                       DEPENDING ON VALUE OF          * 00003500
           C *                                       AZ MOD 90.                     * 00003600
```

Example 6-8
Listing of Electives and
Their Uses

Elective:

```
2-5    C *   SMOOTH    17-20    I   NUMBER OF INITIAL SMOOTHINGS*  00003700
2-6    C *   REPEAT    24       I   0 = READ NEW DATA SET        *  00003800
       C *                          1 = USE PREVIOUSLY READ DATA *  00003900
       C *                          AND SYMBOL POINTS            *  00004000
2-7    C *                          AND LEGENDS                  *  00004100
       C *   LNINT     25-28    I   PLOT EVERY 'LNINT' LINE (0=1)*  00004200
2-8    C *   NOZERO    32       I   0 TO PLOT NON STUDY AREA     *  00004300
       C *                            AS ZERO LINES              *  00004400
       C *                          1 TO DELETE NON STUDY AREA   *  00004500
       C *                          2 TO DARKEN NON STUDY AREA   *  00004600
       C *                            SYMBOL 14 (NORTH ARROW)    *  00004700
       C *                            WILL BE BLANK              *  00004800
       C *                          3 TO DARKEN NONSTUDY AREA    *  00004900
       C *                            SYMBOL 14 (NORTH ARROW)    *  00005000
       C *                            WILL BE DARK               *  00005100
2-9    C *   SQRTT     36       I   1 TO TAKE THE SQUARE ROOT OF *  00005200
       C *                             ALL VALUES                *  00005300
2-10   C *   SAMEH     40       I   1 TO USE SAME Z SCALE FACTOR *  00005400
       C *                             AS THE PRECEEDING PLOT    *  00005500
2-11   C *   ENDLIN    44       I   1 TO DELETE ENDLINES         *  00005600
2-12   C *   SYMAP     45-48    I   SYMAP TAPE INPUT(6,5,10 L/'')*  00005700

       C *                          OR THE NUMBER OF LINES/INCH I*  00005800
       C *                          DIFFERENT FROM THE HORIZONTAL*  00005900
       C *                          SCALE. 0 TO INDICATE EQUAL   *  00006000
       C *                          SCALES                       *  00006100
2-13   C *   LEG       50       I   1 TO DELETE CONTROL          *  00006200
       C *                              INFORMATION FROM PLOT    *  00006300
2-14   C *   NSCAL     52       I   1 TO INCLUDE SCALE LEGEND    *  00006400
2-15   C *   FLP       56       I   1 TO INCLUDE A FLOATING PLANE*  00006500
2-16   C *   BASE      60       I   0 TO INCLUDE A BASE          *  00006600
       C *                          1 TO DELETE THE BASE         *  00006700
       C *                          2 TO INCLUDE HIDDEN PART OF  *  00006800
       C *                            BASE                       *  00006900
       C *                          3 TO PLACE ON BLOCK          *  00007000
2-17   C *   PLTDOT    61-64    I   NUMBER OF SYMBOL POINTS      *  00007100
       C *                            (SEE BELOW)                *  00007200
2-18   C *   FDATA     65-68    I   0 DATA IS TO BE READ BY USER *  00007300
       C *                            WRITTEN SUBROUTINE         *  00007400
       C *                          N DATA IS ON FILE N OF TAPE  *  00007500
```

Example 6-8 (Contd)

```
Elective:
        C *                                                 IN STANDARD FORMAT          * 00007600
2-19    C *      NAMIN       25          I                1 TO READ IN AMIN - VALUES     * 00007700
        C *                                                 BELOW AMIN SET TO AMIN       * 00007800
2-20    C *      NAMAX       26          I                1 TO READ IN AMAX - VALUES     * 00007900
        C *                                                 ABOVE AMAX SET TO AMAX       * 00008000
        C *                                               2 TO READ IN AMAX - VALUES     * 00008100
        C *                                                 ABOVE AMAX NOT CHANGED       * 00008200
2-21    C *      IFORM       77-78       I                USER ASSIGNED VALUE WHICH MAY* 00008300
        C *                                               BE USED IN SUBROUTINE DATA     * 00008400
2-22    C *      NLEG        79-80       I                NUMBER OF TEXT LEGENDS         * 00008500
        C *                                                 MAXIMUM OF 50  (SEE BELOW)   * 00008600
        C *                                                                              * 00008700
        C * CARD 3 - CONTROL CARD (2 OF 2)              - DECIMAL                        * 00008800
3-1     C *    VARIABLE   COLUMNS   TYPE                          DESCRIPTION            * 00008900
        C *      ALTITUDE    1-5         F                HEIGHT ABOVE HORIZONTAL PLANE* 00009000
        C *                                                  FOR VIEWING POSITION ( IN* 00009100
        C *                                                  DEGREES)                    * 00009200
3-2     C *      AZIMUTH     6-10        F                AZIMUTH IN DEGREES             * 00009300
3-3     C *      WIDTH       11-15       F                WIDTH IN INCHES OF FRONT       * 00009400
3-4     C *      HEIGHT      16-25       F                HEIGHT IN INCHES               * 00009500
        C *                                                 WIDTH+HEIGHT (WHICH ARE      * 00009600
        C *                                                 SHORTENED SLIGHTLY           * 00009700
        C *                                                 DEPENDING ON THE VIEWING     * 00009800
        C *                                                 ANGLES) MUST BE LESS THAN    * 00009900
        C *                                                 11 INCHES                    * 00010000
3-5     C *      AMIN        26-35       F                SPECIFIED MAX/MIN OF DATA-IF * 00010100
3-6     C *      AMAX        36-45       F                  BLANK, ACTUAL MAX AND/OR     * 00010200
        C *                                                  MIN WILL BE USED            * 00010300
        C *                                                 WILL BE USED ONLY IF NAMIN/* 00010400
        C *                                                 NAMAX IS NON-ZERO            * 00010500
3-7     C *      VD          46-50       F                VIEWING DISTANCE FOR           * 00010600
        C *                                                TWO-POINT PERSPECTIVES        * 00010700
3-8     C *      STEP        51-55       F                TO USE NON-STANDARD PLOTTER    * 00010800
        C *                                                 STEP SIZE                    * 00010900
3-9     C *      SPBSM       56-60       F                TO USE NON STANDARD PLOTTER    * 00011000
        C *                                                 SYMBOL SIZE                  * 00011100
3-10    C *      SIZE        76-80       F                SIZE OF BLOCK WHEN BASE-3.,    * 00011200
        C *                                               STANDARD = .5 INCH.            * 00011300
        C *                                               MAX = .75 INCH                 * 00011400
        C *                                                                              * 00011500
        C * OPTIONAL DATA CARDS                                                          * 00011600
        C * ......................                                                   * 00011700
```

Example 6-8 (Contd)

```
C *
C * IF DATA VALUES ARE TO BE READ FROM CARDS BY USER SUBROUTINE.           *
C *   THEY SHOULD BE INSERTED HERE.                                         *
C *                                                                         *
C * .....                                                                   *
C *                                                                         *
C * SYMBOL POINTS  ONE CARD FOR EACH POINT PLOTTED. TOTAL NUMBER OF         *
C *   CARDS = 'PLOTDAT'                                                     *
C * .....                                                                   *
C *      ROW         1-10        I                                          *
C *      COL         11-20       I                                          *
C *      SYMBOL      21-30       I          2,3,11,14, OR 21 (14 WILL BE    *
C *                                         TWICE AS LARGE AS OTHERS-       *
C *                                         USE FOR DATA POINTS -           *
C *                                         14 WILL BE BLANK WHEN           *
C *                                         NOZERO = 2                      *
C *                                         14 WILL BE DARK WHEN            *
C *                                         NOZERO=3                        *
C * ..........                                                              *
C * LEGENDS   ONE CARD FOR EACH LEGEND TO BE PLOTTED. TOTAL NUMBER OF*
C *   CARDS = 'NLEG'                                                        *
C * .....                                                                   *
C *      ROW         1-10        I                                          *
C *      COL         11-20       I                                          *
C *      TEXT        30-54       A          (LEFT JUSTIFIED)                *
C *                                                                         *
C *................................                                         *
C *                                                                         *
C *   USER INPUT SUBROUTINE DESCRIPTION...                                  *
C *        SUBROUTINE DATA (X,IR,IC,IFORM)                                  *
C *        DIMENSION X(130)                                                 *
C *                                                                         *
C *   CALLING LIST ARGUMENTS                                                *
C *        X(130)               RETURNS ONE ROW OF DATA MATRIX              *
C *        IR                   NUMBER OF ROW  TO BE RETURNED               *
C *        IC                   NUMBER OF COLUMNS/ROW                       *
C *        IFORM                USER SUPPLIED VALUE TO IDENTIFY DATA SET*
C *                                                                         *
C *   NOTE - IR,IC,AND IFORM SHOULD NOT BE ALTERED.  THEY ARE               *
C *   INCLUDED ONLY AS INFORMATION TO BE USED IN THE SUBROUTINE.            *
C *                                                                         *
C *   THE SUBROUTINE IS CALLED 'ROWS'TIMES(CARD2,COL1-4).  ONE ROW          *
C *   OF THE INPUT MATRIX IS RETURNED EACH TIME.                            *
C *                                                                         *
C ***************************************************************************
```

Example 6-8 (Contd)

6. 7. TECHNICAL INFORMATION

This section is intended to provide technical information to individuals who wish to implement SYMVU-I programs at their local computer installation. As such, this section assumes a working knowledge of your local computer center's systems procedures.

The examples of the job control language were tested on an IBM 360/65 OS MVT release 18 system. SYMVU will compile successfully in FORTRAN IV G and H.

6. 7. 1. SYMVU Source Tape Description

The SYMVU tape is a nine-track tape written at 800BPI with odd parity.

The first file is written in EBCDIC mode with 80 character records blocked 8000 characters per block--i. e., 100 records to a block, and contains approximately 16 blocks. Records 0000100 through 00130300 comprise the SYMVU Fortran IV source program. The records following the source program are sample data which can be used to produce a plot identical to Example 6-3. These records, 0013990 through 00144500, should be punched onto cards and read in as input data to SYMVU at execute time. Example 6-9 shows the job control language (JCL) for punching these source cards from the SYMVU tape. If you are not using the values on the second file that were generated by SYMAP, you will need to use Sub-Routine DATA to read in your matrix of values from cards; these cards should be placed after Elective card 3.

The second file of the tape is a binary set of values which is output from SYMAP. The record form is variable spanned block, with a record length of 524 and a block size of 15724. This file is to be used with the sample data on the first file.

Example 6-9

Utility Program

To Punch Source Cards From SYMVU Tape

```
         111111111122222222223333333333444444444455555555556
123456789012345678901234567890123456789012345678901234567890

//PUNCH EXEC PGM=IEBGENER

//SYSPRINT DD SYSOUT=A

//SYSIN DD DUMMY

//SYSUT1 DD DSN=X,UNIT=TAPE9T,DISP=OLD,VOL=SER=SYMVU1,

// DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DEN=2),LABEL=(1,NL)

//SYSUT2 DD SYSOUT=B

/*

//
```

```
//LKED      EXEC PGM=IEWL,PARM=(XREF,LIST,LET,NCAL),REGION=150K
//SYSPRINT DD   SYSOUT=A
//SYSLIB DD   DSNAME=SYS1.FORTLIB,DISP=SHR
//SYSLIN DD   DDNAME=SYSIN
//SYSLMOD  DD DSN=A4H4405.SYMVU.D,UNIT=SYSLIB,DISP=OLD,
//               VOLUME=(PRIVATE,SER=SWHITE)
//SYSUT1 DD   UNIT=SYSOUF,SPACE=(TRK,(30)),SEP=(SYSLIB,SYSLMOD),          *
//                DSNAME=&&DISKA
//SYSIN DD *,DCB=BUFNO=1
```

SYMVU object deck

```
 NAME SYMVU
/*
//                                                                                    //
```

Example 6-10
Listing of Job Control Language to
Load Binary Deck of SYMVU onto Disk

Example 6-10 shows the JCL for loading the binary deck of SYMVU onto disk.

6.7.2. Minimum Machine Configuration

The machine requirements for SYMVU are:

-- an IBM 360 computer, full operating system;

-- 220K bytes of core storage (unoverlaid SYMVU with a G-level Fortran IV compiler), or, 200K bytes (unoverlaid with an H-level compiler);

-- one disk drive, or three tape drives;

-- a seven-track tape drive for the CalComp plotter; and

-- an eleven-inch CalComp drum plotter.

6.7.3 Units Description

The SYMVU program has statements which reference the FORTRAN IV logical units 1, 2, 3, 5, 6, and 8. These units are referenced in four kinds of FORTRAN statements:

-- READ/WRITE (*) X, Y, Z for binary reads and writes;

-- READ/WRITE (*,1040) for EBCDIC reads and writes;

-- REWIND *; and,

-- ENDFILE *.

(the astrisk, *, refers to the unit number).

The units are used as follows:

(i) Units 1, 2 and 3 are used only for intermediate storage, and are written, end filed, rewound, and read; unit 1 is used to store the input data (the Z-values), unit 2, the symbol points, and unit 3, the legends.

(ii) Unit 5 is used for the card input;

(iii) Unit 6 is used for the printed output describing the plots; and,

(iv) Unit 8 is used for the binary input tape generated by SYMAP; this is the tape of the Z-values which is produced when elective 21 is used.

The CalComp routines will reference another seven-track tape, PLOTTAPE. The program is set up for the 11 inch plotter. By inserting CALL FACTOR (FACT) as statement number 00029250, one can increase the size of the plot output by a specified factor.

6.7.4. Input

The binary tape created by SYMAP contains a description of the grid as its first record. This first record consists of the integer number of rows in the grid, the integer number of columns, the real number of rows per coordinate unit, the real number of columns per coordinate unit, the real minimum value, and the real maximum value. The rest of the tape contains 1 row of data per record. For example, if the grid is 130x118, there will be 130 records of 118 four-byte-reals per record.

The user must use SUBROUTINE DATA if his data is not in this form. A dummy SUBROUTINE DATA is included on the source tape, but the user is expected to provide his own if the format of his data is not comparable to the SYMAP output tape.

Example 6-11 shows the input required to produce the SYMVU in Example 6-3. Example 6-12 shows the catalogued procedure used for SYMVU at the Harvard Computing Center.

6.7.5. Calcomp Plotter Routines Used in SYMVU

PLOTS (BUFFER, 1024)

--- BUFFER is a work buffer for the plot routines.

--- 1024 is the length of BUFFER in words.

--- the pen origin is set at the real bottom edge of the paper.

--- check with your own installation to determine if this is necessary.

LIMIT (3000, & 260)

--- 3000 is set as the number of plotter tape records not to be exceeded.

--- 260 is the Fortran Statement number to branch to if the plot routine attempts to write more than 3000 records.

--- the user may change 3000 or delete the call to this routine altogether depending on his own needs (this routine acts as a protection in case the program attempts to plot a bad point).

PLOT (X, Y, ± IPEN)

--- (x, y) are the coordinates, in inches, of the position to which the pen is to be moved from the current reference point.

--- IPEN=2 to have the pen down during movement; IPEN=3 for pen up during movement; for IPEN=-2 or -3, a new origin is defined at the terminal position after the movement is completed --otherwise, it is as if IPEN were positive

--- if IPEN=999, the effects are the same as if IPEN=3 except that a search record with search address 999 is written and the output device is closed

SYMBOL (X, Y, HEIGHT, IBCD, ANGLE, +NCHAR)

--- (x, y) are the coordinates in inches, of the lower left corner of the first character to be produced; HEIGHT, in inches, of the character to be plotted; IBCD is the text; ANGLE is the angle in degrees measured from the X-Axis; NCHAR is the number of characters to be plotted from IBCD.

NUMBER (X, Y, HEIGHT, FPN, ANGLE, ±NDEC)

--- X, Y, HEIGHT, and ANGLE are the same as they are for SYMBOL.

--- FPN is the floating point number that is to be converted and plotted; if NDEC=0, the number's integer portion and decimal point are plotted, after rounding; if NDEC=-1, only the integer portion is plotted after rounding, if $NDEC < -1$, $|NDEC| - 1$ digits are truncated from the integer position, after rounding.

OFFSET (XOFF, XFAC, YOFF, YFAC)

--- Enters factors to be used by the PLOT routine when IPEN is 2 or 3. See the discussion of FACTOR under the units description.

(For a more complete understanding of the Calcomp routines used in SYMVU, see the Calcomp manual, Programming Calcomp Pen Plotters, California Computer Products, Inc., 305 Muller Avenue, Anaheim, California 92803.

```
//FORT EXEC PGM=G,REGION=150K,PARM='DECK'
//SYSPRINT DD SYSOUT=A
//SYSPUNCH DD SYSOUT=B
//SYSLIN DD DSN=&LOADSET,DISP=(MOD,PASS),UNIT=SYSQUE,
//  SPACE=(TRK,(1,3)),DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200)
//FORT.SYSIN DD *
```

SYMVU Fortran deck that was punched from tape cards 00000100 through 00130300

```
/*
//LKED EXEC PGM=IEWL,PARM='XREF,LIST,LET',REGION=150K,COND=(4,LT)
//SYSPRINT DD SYSOUT=A
//SYSLIB DD DSN=SYS1.FORTLIB,DISP=SHR
//   DD DSN=SYS2.FORTLIB,DISP=SHR
//SYSLIN DD DSN=&LOADSET,DISP=(OLD,DELETE)
//   DD DDNAME=SYSIN
//SYSLMOD DD DSN=&&GOSET(MAIN),UNIT=SYSQUE,SPACE=(1024,(50,20,1)),
// DISP=(MOD,PASS)
//SYSUT1 DD UNIT=SYSQUE,SEP=(SYSLIN,SYSLMOD),SPACE=(1024,(50,20)),
// DSN=&DISKA
/*
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT),REGION=250K
//GO.FT01F001 DD DSN=&SYMV1,SPACE=(TRK,(02,1)),DISP=(,DELETE),
// UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=524,BLKSIZE=15724,BUFNO=1)
//GO.FT02F001 DD DSN=&SYMV2,SPACE=(TRK,(02,1)),DISP=(,DELETE),
// UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=1204,BLKSIZE=1208,BUFNO=1)
//GO.FT03F001 DD DSN=&SYMV3,SPACE=(TRK,(02,1)),DISP=(,DELETE),
// UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=1604,BLKSIZE=1608,BUFNO=1)
//GO.FT05F001 DD DDNAME=SYSIN
//GO.FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=132,BLKSIZE=7260)
//GO.FT07F001 DD SYSOUT=B,SPACE=(80,(39,117),RLSE,CONTIG)
```

Example 6-11
Listing of Job Control Language to run
SYMVU sample input

```
//PLOTTAPE DD DSN=PLOT1,UNIT=TAPE7T,VOL=SER=CALCTP,DISP=(NEW,KEEP),
// LABEL=(,BLP),DCB=DEN=1
//GO.FT08F001 DD DSN=F001,UNIT=(2400,,DEFER),DISP=(,PASS),
// VOL=(,RETAIN,SER=LCGVU1),LABEL=(2,NL),DCB=(RECFM=VSB,LRECL=524,
// BLKSIZE=15724,EROPT=ACC,DEN=2,BUFNO=1)
//GO.FT08F002 DD DSN=F002,LABEL=(3,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F001,DCB=*.FT08F001
//GO.FT08F003 DD DSN=F002,LABEL=(4,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F002,DCB=*.FT08F001
//GO.FT08F004 DD DSN=F002,LABEL=(5,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F003,DCB=*.FT08F001
//GO.FT08F005 DD DSN=F002,LABEL=(6,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F004,DCB=*.FT08F001
//GO.FT08F006 DD DSN=F002,LABEL=(7,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F005,DCB=*.FT08F001
//GO.FT08F007 DD DSN=F002,LABEL=(8,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F006,DCB=*.FT08F001
//GO.FT08F008 DD DSN=F002,LABEL=(9,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F007,DCB=*.FT08F001
//GO.FT08F009 DD DSN=F009,LABEL=(10,NL),DISP=(OLD,PASS),
// VOL=REF=*.FT08F008,DCB=*.FT08F001
//GO.SYSUDUMP DD SYSOUT=A
//GO.SYSIN DD *
FIGURE 8:  MANTEGNA BAY - CONTOUR                                                         00139990
  63 124       4       0       2                      0    1        3  45    1
  45.231.3  6.0         2.                                        .04
          5        110         14                                                         00140000
          6        109         14                                                         00140100
          6        110         14                                                         00140200
```

Example 6-11 (Contd)

```
6     111     14
7     108     14                                        00140300
7     109     14                                        00140400
7     110     14                                        00140500
7     111     14                                        00140600
7     112     14                                        00140700
8     107     14                                        00140800
8     108     14                                        00140900
8     109     14                                        00141000
8     110     14                                        00141100
8     111     14                                        00141200
8     112     14                                        00141300
8     113     14                                        00141400
9     106     14                                        00141500
9     107     14                                        00141600
9     108     14                                        00141700
9     109     14                                        00141800
9     110     14                                        00141900
9     111     14                                        00142000
9     112     14                                        00142100
9     113     14                                        00142200
```

Example 6-11 (Contd)

```
         9     114     14                          00142300
        10     109     14                          00142400
        10     110     14                          00142500
        10     111     14                          00142600
        11     109     14                          00142700
        11     110     14                          00142800
        11     111     14                          00142900
        11      50     14                          00143000
        16      76     14                          00143100
        18      26     14                          00143200
        20      51     14                          00143300
        21     103     14                          00143400
        25      41     14                          00143500
        30      69     14                          00143600
        31      15     14                          00143700
        31      92     14                          00143800
        36      36     14                          00143900
        41      52     14                          00144000
        43      82     14                          00144100
        44      66     14                          00144200
        48      29     14                          00144300
        48     102     14                          00144400
          OS END OF FILE CARD                      00144500
/*                                                   //
//                                                     0
```

Example 6-11 (Contd)

```
//   EXEC  SYMVU,LEVEL=D,TAPE=HU0808,REGION.GO=200K
XX   PROC  LEVEL=A,L=NL,TAPE=SCRTCH,BLOCK=15724,BPI=2                     00000000
XXFORT  EXEC  PGM=G,REGION=150K                                           00000010
XXSYSPRINT  DD  SYSOUT=A,DCB=(LRECL=120,RECFM=FBA,BLKSIZE=7200)           00000020
XXSYSPUNCH  DD  SYSOUT=B,SPACE=(80,(200,50),RLSE),DCB=(LRECL=80,         X00000030
XX   RECFM=FB,BLKSIZE=3200)                                               00000040
XXSYSLIN  DD  DSN=&DATA,UNIT=SYSQUE,SPACE=(80,(200,50),RLSE),             00000050
XX        DISP=(,PASS),DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200)               00000060

//FORT.SYSIN DD *

0001              SUBROUTINE DATA
0002              RETURN
0003              END

XXGO  EXEC  PGM=LOADER,PARM='LET,EP=MAIN',REGION=250K                     00000070
XXSYSLIB  DD  DSN=SYS1.FORTLIB,DISP=SHR                                   00000080
XXSYSLOUT  DD  SYSOUT=A,DCB=(LRECL=121,RECFM=FBA,BLKSIZE=3509)            00000090
XXSYSLIN  DD  DSN=&DATA,DISP=(OLD,DELETE)                                 00000100
XX DD  DSN=A484405.SYMVU.&LEVEL.(SYMVU),DISP=SHR,UNIT=SYSLIB,             00000110
XX VOL=(PRIVATE,SER=SWHITE)                                               00000120
XX    DD  DDNAME=SYSLOAD                                                  00000130
XXFT01F001  DD  DSNAME=&SYMV1,SPACE=(TRK,(02,1)),DISP=(,DELETE),         X00000140
XX UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=524,BLKSIZE=15724,BUFNO=1)            00000150
XXFT02F001  DD  DSNAME=&SYMV2,SPACE=(TRK,(02,1)),DISP=(,DELETE),         X00000160
XX UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=1204,BLKSIZE=1208,BUFNO=1)            00000170
XXFT03F001  DD  DSNAME=&SYMV3,SPACE=(TRK,(02,1)),DISP=(,DELETE),         X00000180
XX UNIT=SYSQUE,DCB=(RECFM=VSB,LRECL=1604,BLKSIZE=1608,BUFNO=1)            00000190
XXFT05F001  DD  DDNAME=SYSIN                                              00000200
XXFT06F001  DD  SYSOUT=A                                                  00000210
```

Example 6-12
Harvard's Catalogued Procedure for SYMVU

```
XXFT07F001 DD  SYSOUT=B,SPACE=(80,(39,117),RLSE,CONTIG)                 00000220
XXPLOTTAPE DD DSN=PLOT1,UNIT=TAPE7T,VOL=SER=CALCTP,DISP=(NEW,KEEP),    X00000230
XX LABEL=(,BLP),DCB=DEN=1                                               00000240
XXFT08F001 DD DSN=F001,UNIT=(2400,,DEFER),DISP=(,PASS),                X00000250
XX VOL=(,RETAIN,SER=&TAPE),LABEL=(1,&L),DCB=(RECFM=VSB,LRECL=524,       00000260
XX   BLKSIZE=&BLOCK,EROPT=ACC,DEN=&BPI,BUFNO=1)                         00000270
XXFT08F002 DD DSN=F002,LABEL=(2,&L),DISP=(OLD,PASS),                   X00000280
XX            VOLUME=REF=*.FT08F001,DCB=*.FT08F001                      00000290
XXFT08F003 DD DSN=F003,LABEL=(3,&L),DISP=(OLD,PASS),                   X00000300
XX            VOLUME=REF=*.FT08F002,DCB=*.FT08F001                      00000310
XXFT08F004 DD DSN=F004,LABEL=(4,&L),DISP=(OLD,PASS),                   X00000320
XX            VOLUME=REF=*.FT08F003,DCB=*.FT08F001                      00000330
XXFT08F005 DD DSN=F005,LABEL=(5,&L),DISP=(OLD,PASS),                   X00000340
XX            VOLUME=REF=*.FT08F004,DCB=*.FT08F001                      00000350
XXFT08F006 DD DSN=F006,LABEL=(6,&L),DISP=(OLD,PASS),                   X00000360
XX            VOLUME=REF=*.FT08F005,DCB=*.FT08F001                      00000370
XXFT08F007 DD DSN=F007,LABEL=(7,&L),DISP=(OLD,PASS),                   X00000380
XX            VOLUME=REF=*.FT08F006,DCB=*.FT08F001                      00000390
XXFT08F008 DD DSN=F008,LABEL=(8,&L),DISP=(OLD,PASS),                   X00000400
XX            VOLUME=REF=*.FT08F007,DCB=*.FT08F001                      00000410
XXFT08F009 DD DSN=F009,LABEL=(9,&L),DISP=(OLD,PASS),                   X00000420
XX            VOLUME=REF=*.FT08F008,DCB=*.FT08F001                      00000430
//GO.SYSUDUMP  DD  SYSOUT=A
//GO.SYSIN DD *
```

Example 6-12 (Contd)

7. THE "GRID" PROGRAM

7.1. INTRODUCTION

GRID is a computer program which has been created to provide a highly efficient means for graphic display of information collected on the basis of a rectangular coordinate grid. The "GRID" Program is designed for use by persons with very little programming experience. However, it is usually necessary for the user to specify his own data formats in Subroutine FLEXIN and this requires an elementary knowledge of FORTRAN IV.

The program is written in FORTRAN IV and is currently being operated on an I. B. M. 360/65 computer at Harvard using 150K bytes. With small programming changes it can be operated on the I. B. M. 7094 with 32K memory. It is possible to operate this program on a smaller machine--with a memory of at least 12K words.

7.1.1. Basic Principles

Each data value is assumed to be associated with a cell on a grid. It is essential that the values should be processed in the correct order, since the program accepts the data in the order in which it prints the map. By the standardized printing process, the program starts at the top of the map and processes the data horizontally row by row and from left to right in each row. Thus the numbers below represent the order in which thrity data values on a 6 x 5 grid will be processed and printed.

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30

The size and shape of the user's grid must be specified using Elective 1.

In the mapping process, the actual data values are generalized into groups, each group having a unique graphic symbol associated with it. The groups into which data are to be placed and the associated symbols may be specified by the user, using Electives 3 through 6. Two types of symbolism are available: a grey scale of symbolism from light-dark which must be specified by the user, or a dot map in half inch square cells. The coordinates of each grid cell location may also be printed. Most of the SYMAP electives for scale generalization are available on GRID.

7.1.2. Required Input

To obtain a graphic display (or map), the user must provide three sets of instructions and has the option of providing a fourth set. The instructions are prepared in the following packages: Data Package (usually a separate tape), Map Package, Irregular Outlines Package (Optional), and Subroutine Flexin.

(i) The Data Package contains the data or numermical information which generates the graphic display. The program is written for a maximum of 10,000 grid cells, but the Multiple Data Set Option permits the user to handle unlimited numbers of data cells.

(ii) The Map Package permits the user to specify the precise form of the map output in terms of various electives.

(iii) The Irregular Outline Option allows the user to specify the boundaries of the study area, if he is dealing with a grid which is not rectangularly bounded.

(iv) Subroutine Flexin is a fortran subroutine which allows the user to specify the format of the DATA. (In a special case described under Data Input Section, it is not necessary to provide this subroutine).

In the following sections, first the content and then the format of each set of inputs is described. At the end there is a description of the organization of a complete submission for the computer.

7.2. DATA PACKAGE

The GRID program provides two separate and distinct input procedures for the data. These are designed to optimize the efficient use of the program under two different types of operation.

7.2.1. Data Option A

Option A uses GRID as an independent program, in which Subroutine FLEXIN is used (i) to read the data from any file format and, (ii) to perform relatively simple statistical calculations on the data so as to generate the value to be mapped. In this case the data is processed one cell at a time. This option permits a wide range of flexibility in the organization of the data.

It is recommended that people who have limited experience with the computer use this option initially. The section on Subroutine FLEXIN describes the details of the procedures in detail. A data file often contains many different variables referring to each grid cell. As there is not enough memory space available to store each variable, it is necessary to read the basic data file each time a map is made. Since it is inefficient with large sets of data to submit a card file for each separate map, it is recommended that such files be prestored on a disc or tape. This permits the user to rewind the files between maps.

The GRID program will automatically rewind a file being read on Fortran Logical I/O unit 12 between each map.

This option is activated by specifying a number greater than zero in field 1 of Elective 2 (see Map Package). The user is cautioned to note the further optional use of this number as described under Elective 2 and Subroutine FLEXIN.

7.2.2. Data Option B

Option B uses GRID as the final 'job step' in a series of 'job steps' made up of complex statistical manipulation routines. In this case the data used to create the graphic display is transferred to the GRID program as a series of Binary arrays with one array (or logical record) for each row of the map. The program will expect one real value in the array for each cell in a row (Note that if an irregular outline is used, there will be variable length records).

The data will only be accepted in this mode under this option. This option is used automatically if Elective 2 is not specified; it is also used if zero is specified in field 1 of Elective 2.

When using this option the data is transferred to GRID using Fortran Logical I/O unit number 20.

7.2.3. Assigning Data to Map Levels

The GRID program internally assigns the value for each grid cell to a level or group. The maximum number of levels is 10, numbered 0-9. If the data is already prescaled such that it has integer values from 0 thru 9, this section of the program may be bypassed by using Elective 14. When large numbers of grid cells are involved, there may be significant time savings.

When the dot map option is used the maximum number of levels is 20, numbered 0-19.

7.2.4. Multiple Data Sets

In its standard form this program is limited to 10,000 data cells. On the IBM 7094 there is, in fact, memory space for about 15,000 data cells. On smaller machines there may be space for less than 10,000 cells. An experienced programmer may adjust this limit by changing the size of "COMMON P (10,000)".

This program was written to handle nearly unlimited amounts of data. An internal loop has been built into the program which permits the user to map as many data cells as necessary on one map. In order to do this, larger data files should be divided into "sets" which have less than 10,000 cells each. The data sets should each refer to a basic grid which is the same size and shape.

When multiple data sets are used, each set is processed separately. Therefore, it is essential that the maximum and minimum of the data ranges (electives 4 and 5) be specified, as there is no guarantee that all data sets will have the same value range. When using multiple data sets, specify the number of sets to be mapped in field 3 of elective 2.

If no irregular outline is used, then data sets may be above, below, or beside each other; the order and spatial relationships are not important. However, when an irregular outline is used, the complete outline must be stored prior to the processing of all the data sets. It is suggested that data sets with an irregular outline be organized such that the first set is the top section of the map and the last set the bottom section of the map, with each set being the full width of the map.

Experienced programmers may want to differentiate between each data set during the processing of Subroutine Flexin. A fourth calling "argument" for Flexin is transfered in the "CALL" statement, following the value of FIRST. This argument gives the number of the data set being mapped.

7.3. MAP PACKAGE

This package instructs the computer to make a map based on the data supplied. It is used to specify the precise form of the map in terms of certain available map electives.

It is most important to remember that:

(i) Once an elective has been specified, it will be carried on to successive maps unless it is changed, and,

(ii) Electives 1 and 7 must be included with the first map of a submission as no standard condition is created by the program for these electives.

On the first card of this package, punch "MAP" in columns 1-3. On the last card, punch the number "99999" in columns 1-5.

On the second, third and fourth cards, punch the title you wish to have appear below the map.

Be sure your title is clearly descriptive so as to differentiate the particular map from all other maps of a similar nature which may be run. One or more of these three cards may be left blank if desired, but all three cards must be included.

On the other cards--to be inserted between the fourth and last card--punch any "electives" desired. Whenever a map elective is not specifically called for, the standard result described under each elective will automatically occur. Therefore, use an elective if you wish to procure a result different from the standard.

7.3.1. Standard Format

A standard format is used for the electives with the exception of electives 7, 10, and 13. It is:

(i) the elective number is punched as an integer in columns 4 and 5 (right justified).

(ii) columns 6-10 are left blank

(iii) six fields of ten columns are defined as follows:

Field	Columns
1	11-20
2	21-30
3	31-40
4	41-50
5	51-60
6	61-70

NOTE:

--A "field" is a set of columns. (These are shown on the computer cards to be punched.) The computer expects to read one variable from each field.

--Integer numbers may not contain decimal points. When punching integer numbers in a field, the lowest end of the number (e.g.: if the number is 339, the nine) must be in the right hand column of the field. The number is then right justified.

--The six fields specified for the elective cards are real numbers and should contain a decimal point. They may be located anywhere within the defined columns. The decimal point may be omitted if the number is right justified.

7.3.2 Elective 1: Grid (1 card)

This elective specifies the parameters for the rectangular grid that is to be mapped. In field 1 specify the number of rows of grid cells down the map and in field 2 specify the number of columns of grid cells across the map. In fields 3 and 4 specify the size of each printed grid cell, in terms of the number of characters down (field 3) and across (field 4). (Remember that each character to be printed measures 1/8" down and 1/10" across.) If fields 3 and 4 are left blank the printed cell size will be 4x5 or 1/2" square.

Enter the numbers in all four fields as decimal numbers. The elective number should be entered as an integer, in column 5.

7.3.3. Elective 2: Data (1 card)

This elective controls the input options for the data. To activate Data Option A (See section on the Data package), specify a number greater than zero in field 1. To return to Option B, specify zero in field 1. Field 2 is not used. If the multiple data set option is to be used, specify in field 3 the number of data sets to be mapped. Standard is one data set and data input option B. The number specified in field 1 (for Data Option A) is transferred to Subroutine Flexin as the value for IFORM. The use of IFORM is discussed in the section on Subroutine Flexin.

7.3.4. Elective 3: Number of Levels (1 card)

To specify the number of levels or class intervals into which the total value range is to be divided (from 2 to 10), punch the decimal number desired in field 1. Standard is ten levels.

7.3.5. Elective 4: Value Range Minimum (1 card)

To specify a number to be used as the minimum value of the total value range, punch the decimal number desired in field 1. Standard is to use the minimum value of the data. To return to standard, specify 1.0 in field 2.

7.3.6. Elective 5: Value Range Maximum (1 card)

To specify a number to be used as the maximum value of the total value range, punch the decimal number desired in field 1. Standard is to use the maximum value of the data. To return to this standard, specify 1.0 in field 2.

7.3.7. Elective 6: Value Range Intervals (1 to 2 cards)

This elective controls the value range for each level or interval. The total value range of the data (as modified by the minimum and maximum of Electives 4 and 5) will be divided up into the number of levels specified in Elective 3. Standard is to have each level or interval assigned an equal range. (See Example 7-1)

To specify the desired range for each level, values proportionate to the size of the desired ranges are used. These should be punched as decimal numbers: in field 1 for the level, etc. (See Example 7-2) Only if there are more than 6 levels, continue on a second card, punching the number for the seventh level in field 1, for the eighth level in field 2, etc. (See Example 7-3) There is a maximum of ten levels for grey scale symbolism and twenty levels for a DOT symbolism.

To return to the standard, specify 0.0 in field 1.

7.3.8. Elective 7:
Symbolism (5 cards)

This elective specifies the grey scale symbolism that will be printed on the map. Because no standard symbolism is stored in the program, this elective must be included on the first map of any submission. All five cards must be included each time it is used.

On the first card punch the identifying elective number "7" in column 5.

On the second card punch in the columns listed below the basic characters desired. Any printer characters may be used.

On the third, fourth and fifth cards punch in the columns listed below any overprint characters desired. If no over-printing is desired, these three cards will be blank.

Example 7-1
Equal Value Range Levels

Using 3 levels with values ranging from a minimum of 0 to a maximum of 30,

--the first level will contain values from 0 thru 10,

--the second level will contain values from 10 thru 20, and

--the third level will contain values from 20 thru 30.

If a data value is exactly equal to a level limit (10, 20, or 30 in this case) it will be assigned to the lower level; e.g. 20 would be assigned to the 10 thru 20 level.

Example 7-2
Unequal Value Range Levels
Four Levels

If the data is to be divided into four groups--the lowest 10%, the next 25%, the next 35%, and the remainder--the one card for elective 6 would be punched as follows:

Column 5	11-20	21-30	31-40	41-50
6.	10.	25.	35.	30.

Example 7-3
Unequal Value Range Levels
Seven Levels

To specify these value range intervals --

Level 1	0-150
Level 2	150-200
Level 3	200-271
Level 4	271-500
Level 5	500-750
Level 6	750-889
Level 7	889-1000

--punch the two cards for elective 6 as follows:

Column 5	11-20	21-30	31-40	41-50	51-60	61-70
6	150.	50.	71.	229.	250.	139.
	111.					

Columns 1-10 are used to specify the general symbolism for each level (column 1 for the symbol to designate the first level, etc.--for as many levels as are to be used.)

Columns 11-20 are used to specify the special symbolism for the respective flag points (column 11 for the symbol to designate flag points in the first level, etc.). The flag point is the central character of a grid cell.

Column 21 is used to specify the symbolism for a value less than the minimum specified in Elective 4.

Column 22 is used to specify the flag point symbolism for a low value.

Column 23 is used to specify the symbolism for a value greater than the maximum specified in Elective 5.

Column 24 is used to specify the flag point symbolism for a high value.

Column 25 is used to specify background symbolism--the symbolism to appear outside the outline of the study area.

Example 7-4 shows a grey scale for ten levels of symbolism.

7.3.9. Elective 8:
Flag Point (1 card)

The flag point is the central character of a grid cell. The special symbolism specified in elective 7 is printed at this flag point. To suppress the printing of

Example 7-4
Level Symbol Specification

This example shows:

- -- a grey scale for ten levels of symbolism (columns 1-10)
- -- flag point symbolism for the ten levels (columns 11-20),
- -- blank low value symbolism and flag point (columns 21 and 22),
- -- blank high value symbolism and flag point (columns 23 and 24), and
- -- blank background symbolism (column 25).

Column	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
card 1					7																				
card 2	.	,	,	+	X	O	O	O	O	O	0	1	2	3	4	5	6	7	8	9					
card 3			.	-	X	O	O	O	O	A															
card 4							-	+	X	X															
card 5										V															

To create a grey scale for less than ten levels it is suggested that the level symbols should be eliminated in the following order:

for:	9 levels	eliminate:	2
	8 levels		2, 9
	7 levels		2, 9, 8
	6 levels		2, 9, 8, 3
	5 levels		2, 9, 8, 3, 6

The flag point symbolism should be adjusted accordingly.

special symbolism at the flag point specify 1.0 in field 1. If it is desired to reinstate the flag point in subsequent maps, specify 0.0 in field 1.

When a map is made with a 1 x 1 character grid, the flag point is automatically suppressed and it must be reinstated for subsequent maps. Standard is the special symbolism at the flag point.

7.3.10 Elective 9: Histogram (1 card)

This Elective controls the printing at the bottom of the map. Specify 1.0 in field 1 to generate a histogram bar chart at the bottom of the map. This bar chart shows the frequency of grid cells in each level. Specify 1.0 in field 2 to suppress the numeric information which is printed with levels. Standard is no Bar chart and inclusion of numeric information. To return to the standard, specify 0.0 in the relevant field.

7.3.11. Elective 10: Text (3-32 cards)

If additional explanatory information is desired--beyond that contained in the map title--this elective may be used to print up to 30 lines of text below the map.

On the first card punch the identifying elective number "10" in columns 4-5.

On not more than 30 other cards, to be inserted between the first and last, punch in columns 1-72 any supplementary information likely to be helpful for future reference, such as source and date of data or the name of the person running the map.

On the last card, punch "ENDTEXT" in columns 1-7. Standard is to have no text, but, generally, some explanatory text is desirable.

7.3.12 Elective 11: Data Record (1 card)

If a printout of the data values--before scaling--is desired, punch 1.0 in field 1. If a punched deck of the data values is desired, punch 1.0 in field 2. If a punched deck of the level numbers to which the data has been assigned is desired, punch 1.0 in field 3. The Standard is no print or punch. To return to the standard, specify 0.0 in the relevant field.

7.3.13. Elective 12: Dot Map (1 card)

As an alternative to the normal symbolism, a dot map can be produced using 4 x 5 grid cells, and the symbol Ø. The data range may be divided into twenty levels (19 if a maximum value is specified in elective 5). The number of characters printed in the cell is equal to the number of the level: if the value falls in level 1, only one of the 20 characters is printed, but if it falls in the 20th level, all 20 characters are printed.

This elective supercedes the specifications of grid cell size in elective 1 and the number of levels in elective 3.

To specify DOT symbolism, punch 1.0 in field 1. To reinstate the grey scale symbolism (specified in elective 7) punch 0.0 in field 1. The grey scale symbolism is standard.

7.3.14 Elective 13: Grid Numbering (1 to 2 cards)

This elective generates row and column numbers on all four sides of the grid to assist the user in locating individual cells on the map.

The top left hand cell of the grid is called the Reference Grid Cell (RGC). It provides the coordinates from which all the rows and columns are numbered. If the coordinates of the RGC are not specified, the program assumes them to be:

Column = 1

Row = N

where N is the number of rows specified in elective 1.

In field 1 specify 1.0 for grid numbering. In field 2 specify the column number of the RGC and in field 3 specify the row number. The standard is no numbering. To return to the standard on subsequent maps, specify 0.0 in field 1.

For some specialized uses the basic grid may be subdivided into parts, such as halves or thirds, and a non-continuous numbering system used. For example, a grid may be numbered in the following manner:

	3 4 0 1	3 4 0 4	3 4 0 7	3 4 1 1	3 4 1 4	3 4 1 7	3 4 2 1
3807	.	.	.	.	.	.	.
3804	.	.	.	.	.	.	.
3801	.	.	.	.	.	.	.
3797	.	.	.	.	.	.	.
3794	.	.	.	.	.	.	.
3791	.	.	.	.	.	.	.
3787	.	.	.	.	.	.	.

The basic grid has a R.G.C. of 340,380. There are three subdivisions numbered 1, 4, and 7; this number of subdivisions is specified in field 1. When field 1 contains a number greater than one, the program will read an extra card to find out how to number the subdivisions. Example 7-5 shows how to specify the two cards for this grid.

7.3.15. Elective 14: Prescaled Data (1 card)

This elective bypasses the routine which assigns the data values to levels, as described in the section on the data package. To activate this option, specify 1.0 in field 1. This option will automatically suppress numeric information below the map. The standard is to have the program scale the data.

On subsequent maps, the normal processing is reinstated by specifying 0.0 in field1; the numeric information should also be reinstated using elective 9.

Example 7-5
Grid Numbering
Subdivided Grid

For a basic grid with an R.G.C. of 340, 380 and subdivisions numbered 1, 4, and 7, Elective 13 should be specified as follows:

```
                         1111111111222222222233333333334 -- 8
Column Number   1234567890123456789012345678901234567890     0
-------------------------------------------------------------
Card 1             13            3.0     340.0     380.0
Card 2          147
```

The first card is in the standard format:

-- elective number 13 in columns 4 and 5.

-- number of subdivisions in field 1

-- R.G.C. column and row numbers in fields 2 and 3.

The second card contains the numbers (0-9) for the subdivisions, the first subdivision in column 1, second in column 2, etc.

7.3.16 Elective 15: Timer (1 card)

This elective causes the executive times for various stages of the program to be printed with the map package information. To activate this option, specify 1.0 in field 1. The standard is not to have the times printed.

(at installations other than Harvard the calls to the clock routine may require modifications).

7.3.17. Elective 16: Legends (1 card) (only available in a special version of the program)

This elective causes a legend tape--created by the SYMAP Program--to be read and the legends to be printed on the GRID map.

The standard is not to have legends; normally, when legends are desired, special edited decks of the SYMAP and GRID programs are used to create and read in the legends and elective 16 is not used.

The last card of the map package should have "99999" punched in columns 1-5.

7.4. IRREGULAR OUTLINES

Although the data has been collected on the basis of a regular grid, the outline of the study area may not be rectangular. There are two methods for handling this problem in "GRID":

7.4.1. Filling in the Rectangle.

The program expects to read a data value for each cell. When an irregular outline exists, the user may complete the rectangle with data records (generally, one grid cell data record per card) which indicate that the cell should be printed in the background symbolism. As the data values are read in Subroutine Flexin, the occurrence of the blank or background cells must be checked. When a background cell occurs, it should be assigned a data value of -999999.0. This value activates the background symbolism routine and causes the cell to be printed as background.

The easiest background indicator to code is a zero, or blank, unless zero is a valid value. For simplicity the data value of - 999999.0 can be coded directly.

7.4.2. Irregular Outline Package

To simplify the handling of irregular outlines a small routine has been built into the program so that the user can specify the shape of the outline without filling up the rectangle with data records. The irregular outline is specified in terms of the number of cells from the vertical edges of the grid--left and right borders--that are to be left blank in each row. Background symbolism will automatically be assigned to these cells.

The information is given to the computer in a separate package called the Irregular Outline Package.

It is specified as follows:

On the first card IRREGULAR OUTLINE is punched in columns 1-17.

On the last card 99999 is punched in columns 1-5.

Between the first and last card a series of cards is punched with the following format:

> In columns 1-5 the number of successive rows for which the particular format is repeated;
>
> In columns 6-10 the number of blank cells at the beginning of the row; and,
>
> In columns 11-15 the number of blank cells at the end of the row.

These numbers are integer numbers; they must be right justified and contain no decimal points. Since the program processes these cards in order, the first card refers to the top row (or rows, as specified in columns 1-5), the second card refers to the second row (or, first format change)

Example 7-6 shows how such an irregular outline package would be used.

This package must precede the first Map Package to which it refers. Once it has been entered, it will be used on all successive maps until it is replaced by a new package or deleted by a blank package. A blank package contains only the first and last cards; it restores the rectangular grid as the outline.

This routine is limited in that it can only handle irregularities which are contiguous to a vertical edge of the grid. In some cases--such as indentations at the

Example 7-6
Irregular Outline Package

Assume that we wish to produce a map as in the diagram below. The numbers show the order in which the data values would be read in, if we use an irregular outline package.

(Row)							(Number of Background Cells per row)
1	*	*	*	1	2	3	3
2	*	*	*	4	5	*	4
3	*	*	6	7	*	*	4
4	*	8	9	10	*	*	3
5	11	12	13	14	15	*	1
6	16	17	18	19	20	*	1
7	21	22	23	24	25	26	0
8	*	27	28	29	30	31	1
9	*	32	33	34	35	36	1

The astricks represent the blank cells. Their location would be specified in the irregular outline package as follows:

(Row)	Column 11111111 12345678901234567 IRREGULAR OUTLINE			(Number of Background Cells per row)
1	1	3	0	3
2	1	3	1	4
3	1	2	2	4
4	1	1	2	3
5 and 6	2	0	1	1
7	1	0	0	0
8 and 9	2	1	0	1
	99999			

The information for the nine rows on the map is supplied on only seven cards, since the same formate is used successively in two cases. The numbers in columns 1-5 must add up to the total number of rows, or the program will stop. Every row must be accounted for in this package, even if it has no blank cells.

If the first method of specifying an irregular outline is used--filling in the rectangle, the data values would be read in in the order shown below:

(Row)							(Number of Background Cells per row)
1	(1)	(2)	(3)	4	5	6	3
2	(7)	(8)	(9)	10	11	(12)	4
3	(13)	(14)	15	16	(17)	(18)	4
4	(19)	20	21	22	(23)	(24)	3
5	25	26	27	28	29	(30)	1
6	31	32	33	34	35	(36)	1
7	37	38	39	40	41	42	0
8	(43)	44	45	46	47	48	1
9	(49)	50	51	52	53	54	1

The numbers in brackets would have values which indicate that the cell should be printed in the background symbolism.

top or bottom of the grid or blank areas in the middle of the map--it is necessary to fill in these areas, with background cells as specified in A, the first method for handling irregular outlines.

7.5. SUBROUTINE FLEXIN

FLEXIN is a FORTRAN IV subroutine which is used to specify instructions about the data value to be mapped for each grid cell. These instructions may specify:

(i) where the value to be mapped is located on a data card or in a data file on tape or disk; or,

(ii) what statistical analysis is to be performed on a variable, or variables, to derive the value to be mapped.

This subroutine is called by the main program once for each data cell that is to be mapped. Each time it is called, it reads the data card or file which refers to the data cell. Example 7-7 shows the simplest use of subroutine Flexin: specifying where the value to be mapped is located on a data card. Example 7-8 shows more complicated uses of the Subroutine--where statistical analysis is performed on variables and where data for more than one map must be read in.

These examples of the use of Subroutine Flexin are only intended to demonstrate the utilization of the arguments of the Subroutine (IFORM, T, FIRST). The user who is familar with FORTRAN IV should be able

Example 7-7

Simplest Use of Subroutine Flexin

```
                  1111111111222222222233333333334    8
Columns  1234567890123456789012345678901234567890...0

               SUBROUTINE FLEXIN (IFORM,T,FIRST)
               READ (5,100) T
           100 FORMAT (5X,F5.2)
               RETURN
               END
```

This routine instructs the program to read the variable T from cards (the unit 5 specified in the read statement is the card reader) according to the format found in statement number 100. This format says that the value will be found in columns 6-10 on each card. Each time that the Subroutine is called--once for each data cell--a new value for the variable T will be read and returned to the main program as the value to be mapped for that data cell.

Example 7-8
Multiple Uses of Subroutine Flexin

```
                   1111111111222222222233333333334      Explanatory
Columns   1234567890123456789012345678901234567890        Notes:
                SUBROUTINE FLEXIN (IFORM,T,FIRST)             1
                DIMENSION CONST(10)                           2
                LOGICAL FIRST                                 3
                GOTO (1, 2, 3, 4, 5), IFORM                   4
              1 CONTINUE                                      5
                READ(5,100)AREA,POP
            100 FORMAT (2F5.0)
                T=POP/AREA
                RETURN
              2 CONTINUE                                      6
                IF(.NOT.FIRST)GOTO 21
                READ(5,200) (CONST(I),I=1,10)                 7
            200 FORMAT(10F5.2)
                REWIND 12
             21 READ( 12)I,PICNIC,CAMPER                      8
                T=PICNIC+CAMPER*CONST(I)
                RETURN
              3 CONTINUE
                RETURN
              4 CONTINUE
                RETURN
              5 CONTINUE
                RETURN
                END
```

Example 7-8
Multiple Uses of Subroutine Flexin
(Continued)

NOTES:

(1) The variables IFORM and FIRST are carried into the subroutine as control variables. The data value to be mapped is carried back to the main program as T. (Remember that subroutine Flexin is called once for every value to be mapped.)

(2) Storage space is created for ten values of the array CONST.

(3) The variable FIRST is declared to be a logical variable. It is "true" on the first entry to Flexin and "false" on all other entries from the main program. This is set outside the Subroutine.

(4) As more than one map is to be made and as each requires a different routine to read in the data, the rest of Flexin is broken down into segments named 1, 2, 3, 4, 5. The variable IFORM is given its value for each map in Elective 2, field 1. After reading this statement the program jumps to the statement N CONTINUE where N is the value of IFORM. In the example above there are only two routines, but there is space for three more if needed. By extending the "GOTO" statement, as many routines as needed may be used. Each routine is sandwiched between a CONTINUE statement (which indicates the beginning of each routine) and a RETURN statement (which sends the value T back to the main program).

(5) The first segment is an extension of Example 7-7. In this case two data values are read, AREA and POP(ulation), and from them the population density is calculated and returned to the main program as the value to be mapped.

Example 7-8
(Continued)

(6) The second segment is intended to create a value showing the amount of use a park receives each day. The variable PICNIC is the average number of picnics per day; the variable CAMPER is the average number of campers entering the park each day. Campers tend to stay for different lengths of time in different types of parks, so the number of campers entering daily is weighted for each of ten types of parks.

(7) The ten different weights are read from one data card at the beginning of the data deck on the first entry to Subroutine Flexin. In all successive entries the logical variable FIRST will be "False", and this section will be by-passed: the program will jump from statement number 2 to statement number 21. The logical unit 12 is also rewound on this first call to Flexin. By rewinding the file we ensure that the first record read in statement number 21 is the first data record desired.

(8) In each entry to Flexin the program reads the variable I (the type of park), PICNIC (Number of picnickers) and CAMPER (number of campers). This data is on a disk file mounted on logical Unit 12, rather than on cards. The park type (I) specifies which pre-stored value of CONST is to be used to calculate the value to be mapped.

to devise more sophisticated analysis and statistical routines to be applied to his data.

7.6. COMPUTER SUBMISSIONS

After the packages have been prepared, they must be placed in the correct order together with the control cards needed for submission to the computer. The normal order of a deck of cards for submission to an IBM 360 computer is:

Control Cards
Fortran Program (including Subroutine FLEXIN)
More Control Cards
Data on which the program is to operate

As procedures vary at each installation, the user is advised to consult with a programmer who knows how his own installation handles the control cards and its exact procedures.

The data on which the program operates consists of IRREGULAR OUTLINES PACKAGE, the MAP PACKAGE, and the DATA INPUT. These packages must be in the correct position with respect to each other in the card input deck:

(i) The IRREGULAR OUTLINES package must precede the MAP package to which it refers. Once an IRREGULAR OUTLINE has been specified it will be used for every MAP package until it is supressed.

(ii) Each time the program reads a MAP package it will attempt to make a map. There is no limit to the number of MAP packages in any one submission.

(iii) If the DATA INPUT is being read from cards, it must immediately follow the MAP package to which it refers.

The end of the data input is signaled by a card with "END" punched in columns 1-3 immediately following the last MAP PACKAGE--or the last data card, if the data is on cards. Example 7-9 shows a typical submission where four maps are required, the data input is on cards for one map and tape or disk for the other three, and an irregular outline is specified for two maps but not for the other two.

In each case when the computer encounters a MAP PACKAGE, it will make a map based on the information which preceeds the package plus any DATA INPUT on cards immediately following the package. The information proceeding the package is generally an IRREGULAR OUTLINES PACKAGE; the package will be retained for successive maps until it is changed or removed.

Example 7-9
Sample Submission for Four Maps

```
Control Cards             .
Fortran Programs          .
Control Cards             .
Data                      .  IRREGULAR OUTLINES       .  irregular out-
                             --                          lines for first
                             --                          map
                             --
                             --
                             99999
                             MAP                      .  first map
                             --                          package
                             --
                             --
                             --                       .  irregular out-
Data input for first map  .  99999                       line retained
on tape or disk              MAP                         for second map.
                             --                       .  second map
                             --                          package
                             --
                             --
Data input for second     .  99999                    .  Outline removed
map on tape or disk          IRREGULAR OUTLINE           for third map
                             99999                       with blank package
                             MAP                      .  third map
                             --                          package
                             --
                             --
                             --
Data input for third      .  99999
map on cards                    9.0      17.2
                               10.0      18.1
                              - - - - - - - -
                              - - - - - - - -
                              - - - - - - - -         .  fourth map
                                7.1      20.3            package
                             MAP
                             --
                             --
                             --
Data input for fourth     .  --                       .  indicates end
map on tape or disk          99999                       of data.
                             END
```

8. PROGRAM MODIFICATIONS

A number of significant changes in the SYMAP and GRID programs were made during the project to facilitate the mapping of the coal data, particularly data having to do with quantities of coal. In the SYMAP program changes were made in the printout of the textual information below the map. The major changes to the GRID program involved the rewriting of the legends routine that had been included in previous versions of the program run on the IBM 7094. This enables the same kind of legends used with the SYMAP program to be displayed on a map produced by GRID. Formerly it was not possible to include legends with the GRID program. The use of legends also required the rewriting of a subroutine in SYMAP.

The changes made to both the SYMAP and GRID programs were not incorporated into the existing standard versions of the programs. Instead, the specific changes to the standard programs necessary to duplicate the work of this project are documented in the following sections. The final section provides the necessary user-specifications for the C-LEGENDS package. Since this package is rarely used --it has been replaced by the C-OTOLEGENDS package--the specifications were not covered under the section entitled Preparation of Packages.

8.1 PRINTOUT STATEMENTS IN SYMAP

Two changes were made in the way that textual information following the maps is printed out by the SYMAP program. Since the clock subroutine is presently inoperable for runs on the IBM 360-65 at the Harvard Computing Center, all statements specifying TIME= were deleted from the printout. Secondly, the way in which the ranges of data values assigned to each of the data value levels has been specified in the past has been unclear to many users of the program. Therefore, the lines stating "absolute value range applying to each level (maximum included in highest levels only)", were deleted from the printout.

The explanation of the value ranges specified which follows this statement was changed as follows:

Value range changed from

level	1	2
Minimum	1.00	2.00
Maximum	2.00	3.00 etc.

to

level	1	2
Minimum	1.00	2.00
Maximum	1.99	2.99 etc.

In the normal version of the program, it was not clear which level the value 2.00 would be assigned to; in the new printout of the level ranges, this confusion has been eliminated. Changes to the SYMAP program were necessary in MAIN and Subroutine FINAL. These are shown in Example 8-1. No other changes need be made to the SYMAP source deck and no changes are required in the way the user specifies his input.

```
C                                                                           F0000100
C     PROGRAM TITLE          [illegible]                                    F0000500
C                                                                           F0000600
C     VERSION                5.1[illegible] FOR SYSTEM 360                  F0000700
C                                                                           F0000800
C     DATE                   JULY [illegible].1970                          F0000900
C                                                                           F0001000
C     ORIGINATOR             HOWARD T. FISHER                               F0001100
C                            NORTHWESTERN UNIVERSITY                        F0001200
C                                                                           F0001300
C     PROGRAMMERS            BETTY BENSON. W. G. VALANCE. DONALD            F0001400
C                            SHEPARD. ROBERT RUSSELL. MARION MANOS          F0001500
C                                                                           F0001600
C     PRODUCER               LABORATORY FOR COMPUTER GRAPHICS AND           F0001700
C                                 SPATIAL ANALYSIS                          F0001750
C                            HARVARD CENTER FOR ENVIRONMENTAL DESIGN        F0001800
C                                 STUDIES                                   F0001850
C                            GRADUATE SCHOOL OF DESIGN                      F0001900
C                            HARVARD UNIVERSITY                             F0002000
C                            CAMBRIDGE. MASSACHUSETTS 02138                 F0002100
C                            UNITED STATES OF AMERICA                       F0002200
C                                                                           F0002300
C                                                                           F0002400
      COMMON /FREQ/ NFREQ(13).NLOC(12).KTIME.KTM                            F0002500
      COMMON/LEVINF/NLEV.AMM(2).SIZLEV(10).ZMN.ZMX                          F0002600
      COMMON /C2/ DPMM(2.2).DLMM(2.2).FIGMM(2.2).COVMM(2.2)                 F0002700
      COMMON/SYMBLS/JUMBLE(4.42).LCHAR(4.100).NLCHAR                        F0002800
      COMMON /TITLES/ TITLE(54).NTX.TEXT(540)                               F0002900
      COMMON /CHAR/ XYA(51.3).LISTP(20).NPT.STR1                            F0003000
      COMMON /INVS/ ZINMIN.ZINMAX                                           F0003100
      COMMON /D1/ J2(4).AJ2(4)                                              F0003200
      COMMON /C10/ RF.RG1.RG.RP.RGS.FRR.FC.DELTAX.DELTAY.TOL.DA.DIAG        F0003300
      COMMON /BUG/ NBUG.BDUMP(5)                                            F0003400
      COMMON /DISTM/ FTPDIS.TOPDIS                                          F0003500
      COMMON /VALM/ ZEMIN.FZEMIN.ZEMAX.FZEMAX                               F0003600
      COMMON /NUMBER/NPTS.NVALS.NIND.NBOUN.NFIG.NLEG.NCOV.[illegible].NIDOL.     F0003700
     1  NSLOPS                                                              F0003800
      COMMON /LOCATN/ INEXT.IPTF.IVALF.INDF.INDLF.ISLOPE                    F0003900
      COMMON /MAPDIM/ CHSPAN(2).XYEXT(2.2).SHIFT(2.2).SCALE.[illegible](2)  F0004000
      COMMON /C3/ NXY(2).RXY(2)                                             F0004100
```

Example 8-1
SYMAP Print Out Changes

```
      COMMON /INOC/ INOC(15.15)                                         F0004200
      COMMON /DIMNSN/ NNC(2).ICRCD(2).RCMAX(2)                          F0004300
      COMMON /CONCO/ PRINT.OPT.FACT(3).DIV(3).TOPT.SONS                 F0004400
      COMMON /XPRINT/ IPRINT                                            F0004500
      COMMON COMMON (10000 )                                            F0004550
C                                                                       F0004600
      LOGICAL PRINT.OPT.SONS                                            F0004610
C     WRITE PROGRAM HEADING                                             F0004700
      WRITE (6.1000)                                                    F0004800
C                                                                       F0004900
C     WRITE INITIAL TIME                                                F0005000 .3 deletes
C                                                                       F0005200
C     REWIND INTERMEDIATE TAPES                                         F0005300
      REWIND 1                                                          F0005400
      REWIND 2                                                          F0005500
      REWIND 3                                                          F0005600
      REWIND 4                                                          F0005700
C                                                                       F0006100
C     REPEAT FOLLOWING LOOP ONCE FOR EACH MAP                           F0006200
   10 CONTINUE                                                          F0006300
C                                                                       F0006400
C     CALL INPUT ROUTINE                                                E0006500
      CALL INPUT                                                        E0006600 .a deletion
C                                                                       F0007000
C     CALL INITIALIZATION ROUTINE                                       F0007100
      CALL INIT (SONS)                                                  E0007200 .3 deletes
C     CALL MAPPING ROUTINE                                              F0007600
      CALL MAP                                                          F0007700 .3 deletes
C                                                                       F0008100
C     CALL FINALIZATION ROUTINE                                         F0008200
      CALL FINAL                                                        F0008300 .3 deletes
C                                                                       F0008700
C     GO ON TO NEXT MAP                                                 F0008800
      GO TO 10                                                          F0008900
C                                                                       F0009000
 1000 FORMAT (17H1SYMAP. VERSION 5/56H0LABORATORY FOR COMPUTER GRAPHICS F0009100
     1AND SPATIAL ANALYSIS/48H HARVARD CENTER FOR ENVIRONMENTAL DESIGN SF0009200
     2TUDIES/26H GRADUATE SCHOOL OF DESIGN/19H HARVARD UNIVERSITY/      F0009300
     3 31H CAMBRIDGE. MASSACHUSETTS 02138/25H UNITED STATES OF AMERICA) F0009400
      END                                                               F0009600
```

Example 8-1 Contd

```
      BLOCK DATA                                                        F0009800
C     INITIALIZES MAP NUMBER, TIME, AND [illegible]                     F0009900
C                                                                       F0010000
      COMMON /KEEP/ MAPNO,TIME,NSTOP,ITPCNT                             F0010100
      DATA ITPCNT/0/                                                    F0010200
C ITPCNT COUNTS THE NUMBER OF MAPS WRITTEN ON TAPE 8 UNDER OPTION 31    F0010250
C                                                                       F0010300
      DATA MAPNO/0/,TIME/0.0/,NSTOP/10000/                              F0010400
C                                                                       F0010500
      END                                                               F0010600
      LOGICAL FUNCTION VALID(Z)                                         F0013000
C     RETURNS TRUE IF Z IS OKAY                                         F0013100
C                                                                       F0013200
      COMMON /INVS/ ZINMIN,ZINMAX                                       F0013300
      COMMON /OPTION/ IOPT(40),ITAPER,PCHD                              F0013400
      LOGICAL IOPT,PCHD                                                 F0013500
C                                                                       F0013600
      ZPT=Z                                                             F0013700
      VALID=.NOT. IOPT(18) .OR. ZPT .NE. 0.0 .OR. SIGN(1.0,ZPT) .GT. 0.0F0013800
      IF (.NOT. VALID) RETURN                                           F0013900
      IF (IOPT(19)) GO TO 10                                            F0014000
      IF (IOPT(20)) VALID=ZPT .GT. ZINMAX                               F0014100
      RETURN                                                            F0014200
   10 IF (IOPT(20)) GO TO 20                                            F0014300
      VALID=ZPT .LT. ZINMIN                                             F0014400
      RETURN                                                            F0014500
   20 IF (ZINMAX .GE. ZINMIN) GO TO 30                                  F0014600
      VALID=ZPT .GT. ZINMAX .AND. ZPT .LT. ZINMIN                       F0014700
      RETURN                                                            F0014800
   30 VALID=ZPT .GT. ZINMAX .OR. ZPT .LT. ZINMIN                        F0014900
      RETURN                                                            F0015000
      END                                                               F0015100
      FUNCTION LEVEL(T)                                                 F0015300
C     RETURNS THE LEVEL IN WHICH VALUE T LIES                           F0015400
C                                                                       F0015500
      COMMON /LIMITS/ NLEVEL,ZMAX,ZMIN(10)                              F0015600
C     -1 IS RETURNED IF VALUE T IS HIGH, 0 IF LOW                       F0015700
      IF (T.LT.ZMAX) GO TO 70                                           F0015800
      LEVEL=-1                                                          F0015900
      RETURN                                                            F0016000
```

. Subroutine

Example 8-1 Contd

```
10 DO 20 I=1,[illegible]                                          F0016100
   IF [illegible] .LT. [illegible] GO TO 30                       F0016200
20 CONTINUE                                                       F0016250
   LEVEL=NLEVEL                                                   F0016300
   RETURN                                                         F0016400
30 LEVEL=I-1                                                      F0016500
   RETURN                                                         F0016600
   END                                                            F0016700
```

Example 8-1 Contd

```
      SUBROUTINE [illegible]
C     [illegible] MAIN                                                F0439700
C                                                                     F0439800
      COMMON /PA[illegible]/ [illegible].TIME.NST[illegible].ITLAST   F0439900
      COMMON /OPTION/ IOPT(40).ITAPE                                  F0439910
      DIMENSION ITAPE(13)                                             F0440000
      COMMON /TITLES/ TITLE(54).NTX.TEXT(980)                         F0440001
      COMMON /SYMBOLS/ [illegible](4.30).CCHAR(4.100).NCCHAR          F0440100
      COMMON /ERRORS/ NFREQ(13).NOTES(12).KTIME.KTB                   F0440200
      COMMON /LIMITS/ NLEVEL.ZMAX.ZMIN(10)                            F0440300
      COMMON /NUMBRS/NPTS.NVALS.NIND.NHOON.NFIG.NLFG.NCOV.NCARS.NINDL.F0440400
     1[illegible]                                                     F0440500
      COMMON /LEVELS/ NLEV.AMM(2).SIZLEV(10).ZMN.ZMX                  F0440600
      DIMENSION IFREQ(12).SYM(30.4).ASYM(30.4).SYMBOL(30.4)           F0440700
      DIMENSION VAL(12).HVAL(11)                                      F0440800
      DIMENSION A(120).KY(10).ABOVE(5)                                F0440900
      INTEGER SYM.ASYM.SYMBOL                                         F0441000
      LOGICAL IOPT.FIRST                                              F0441100
      DATA EQSIGN/1H=/                                                F0441200
      DATA KY/2H 1.2H 2.2H 3.2H 4.2H 5.2H 6.2H 7.2H 8.2H 9.2H10/      F0441300
      DATA LOW/2H L/.HIGH/2H H/                                       F0441400
      DATA ABOVE/1HA.1HB.1HO.1HV.1HE/.BLANK/1H /                      F0441500
      DATA A/120*1H /                                                 F0441600
C                                                                              . insert
      IF (NPTS .NE. 0 .OR. NFIG .NE. 0)  GO TO 5                      F0441700
      NFREQ(11)=0                                                     F0441710
      NFREQ(12)=0                                                     F0441720
      NFREQ(13)=0                                                     F0441730
      IF (IOPT(21)) END FILE 8                                        F0441740
C                                                                     F0441750
C     MINIMUM VALUES                                                  F0441800
    5 IF (NFREQ(11) .EQ. 0) GO TO 30                                  F0441900
      J=1                                                             F0442000
      DO 10 I=1.NLEVEL                                                F0442100
      J=J+1                                                           F0442200
   10 VAL(J)=ZMIN(I)                                                  F0442300
      VAL(1)=ZMN                                                      F0442400
      J=NLEVEL+1                                                      F0442500
C                                                                     F0442600
C     PERCENTS FOR EACH LEVEL                                         F0442700
      VAL(J+1) =ZMAX                                                  F0442800
      IF ((ZMAX-ZMIN(1)) .EQ. 0.0)  RANGE=100.00                      F0442900
                                                                      F044[illegible]
```

Example 8-1 Contd

```
      IF ((ZMAX-ZMIN(I)) .NE. 0.0)  RANGE=100.00/(ZMAX-ZMIN(I))          F0443050
      DO 20 I=1,J                                                        F0443100
   20 PVAL(I)=(VAL(I+1)-VAL(I))*RANGE                                    F0443200
      GO TO 60                                                           F0443300
   30 DO 40 I=1,NLEVEL                                                   F0443400
   40 VAL(I)=ZMIN(I)                                                     F0443500
      J=NLEVEL                                                           F0443600
      VAL(J+1)=ZMAX                                                      F0443700
      IF ((ZMAX-ZMIN(1)) .EQ. 0.0)  RANGE=100.00                         F0443800
      IF ((ZMAX-ZMIN(1)) .NE. 0.0)  RANGE=100.00/(ZMAX-ZMIN(1))          F0443850
      DO 50 I=1,J                                                        F0443900
   50 PVAL(I)=(VAL(I+1)-VAL(I))*RANGE                                    F0444000
   60 IF (NLEVEL .EQ. 1)  PVAL (1) = 100.0                               F0444100
      DO 70 I=1,4                                                        F0444200
      DO 70 IJ=1,30                                                      F0444300
   70 SYMBOL(IJ,I)=JUMBLE(I,IJ)                                          F0444400
C                                                                        F0444500
C     TEXT OPTION                                                        F0444600
      IF (.NOT. IOPT(10))  NTX=0                                         F0444700
      WRITE (6,1030)  TITLE                                              F0444800
      IF (NTX .NE. 0) WRITE (6,1040) (TEXT(I),I=1,NTX)                   F0444900
C                                                                        F0445000
C     LOW VALUES NOW BECOME THE FIRST LEVEL                              F0445100
   80 IF (NFREQ(11) .EQ. 0)  GO TO 120                                   F0445200
      J=1                                                                F0445300
      DO 90 L=1,4                                                        F0445400
      ASYM(J,L)=SYMBOL(22,L)                                             F0445500
   90 SYM (J,L) = SYMBOL (21,L)                                          F0445600
      IFREQ (J) = NFREQ (11)                                             F0445700
      DO 110 I =1,NLEVEL                                                 F0445800
      J = J+1                                                            F0445900
      DO 100 L=1,4                                                       F0446000
      ASYM(J,L)=SYMBOL(I+10,L)                                           F0446100
  100 SYM(J,L)=SYMBOL(I,L)                                               F0446200
  110 IFREQ (J)= NFREQ(I)                                                F0446300
      GO TO 150                                                          F0446400
  120 DO  140  I=1,J                                                     F0446500
```

Example 8-1 Contd

```
      DO 130 L=1,4                                                      F0446600
      ASYM(J,L)=SYMBOL(I+1,L)                                           F0446700
  130 SYM(J,L)=SYMBOL(I,L)                                              F0446800
      IF (I [illegible] .EQ. 0) .AND. [illegible] .NE. 0)) [illegible]  F0446810
  140 IFREQ(J)=NFREQ(I)                                                 F0446900
C                                                                       F0447000
C     SET VALUES EQUAL LAST LEVEL                                       F0447100
  150 IF (NFREQ(12) .EQ. 0) GO TO 170                                   F0447200
      J=J+1                                                             F0447300
      DO 160 L=1,4                                                      F0447400
      ASYM(J,L)=SYMBOL(24,L)                                            F0447500
  160 SYM(J,L)=SYMBOL(23,L)                                             F0447600
      IFREQ(J)=NFREQ(12)                                                F0447700
      J=J-1                                                             F0447800
  170 IF (IOPT(24)) GO TO 240                                           F0447900
      WRITE (6,1000) ZMN,ZMX                                            F0448000
 1000 FORMAT (///25H DATA VALUE EXTREMES ARE ,2E12.5///)                F0448100
C                                                                       F0448200
C     TOTAL AMOUNT OF INVALID VALUES                                    F0448300
      IF (NFREQ (13) .NE. 0 ) WRITE (6,1050) NFREQ (13)                 F0448400
      IF (KTIME.GT.0) WRITE(6,1055) KTIME,KTM                           F0448410
 1055 FORMAT (1X,33HTOTAL SUPERIMPOSED DATA POINTS I5,I5,18H. THESE OCCF0448420
     1UR IN ,I3,11H LOCATIONS.//)                                       F0448430
C                                                                       F0448500
C     WRITE OUT MINIMUM MAXIMUM AND PERCENTS                            F0448600
      KWORK=J+1                                                         F0448700
      IF (NFREQ(12) .EQ. 0) GO TO 210                                   F0448800
      NK=(KWORK+10)-                                                    F0449000
      L=NK                                                              F0449100
      DO 180 II=1,NK                                                    F0449200
      JD=II                                                             F0449300
      IF (JD .EQ. L) GO TO 190                                          F0449400
  180 A(II)=BLANK                                                       F0449500
  190 [illegible]                                                       F0449600
      JN=0                                                              F0449700
      DO [illegible]                                                    F0449800
      JNN=JNN+1                                                         F0449900
  200 A(JJ)=ASYM([illegible])                                           F0450000
      GO TO [illegible]                                                 F0450100
  210 [illegible]                                                       [illegible]
```

. a deletion (beside F0448800)

. a substitution (beside the last line)

Example 8-1 Contd

```
                                                                          E0450500
      IF ([illegible](11) .[illegible]. 0 .AND. NEREQ(12) .EQ. 0) WRITE (6,1060)(VAL(I)E0450600
     1,[illegible])                                                       E0450700
      IF ([illegible] .NE. 0 .AND. NEREQ(12) .EQ. 0) WRITE (6,1080)(VAL(I)E0450800
     1,I=1,[illegible])                                                   E0450900
      IF (NEREQ(11) .NE. 0 .AND. NEREQ(12) .NE. 0) WRITE (6,1090)(VAL(I)E0451000
     1,I=1,NWORK)                                                         E0451100
      IF (NEREQ(11) .EQ. 0 .AND. NEREQ(12) .NE. 0) WRITE (6,1050)(VAL(I)E0451200
     1,  ,J)                                                              E0451300
      [illegible] I=NWORK-1                                               E0451310  insert
      [illegible] I=[illegible],NWAV1                                     E0451320
      IF ([illegible] .GT. [illegible]) GO TO 220                         E0451330
      [illegible](I)=VAL(I)-.01                                           E0451340
  220 CONTINUE                                                            E0451350
      WRITE (6,1100) (VAL(I),I=1,NWORK)                                   E0451400
      IF ([illegible] .EQ. 0) GO TO 230                                   E0451410
      IF (NN.GT.120) NN=120                                               E0451420
      WRITE (6,1105) (T(JJ),JJ=1,NN)                                      E0451430  insert
 1105 FORMAT (1H ,9X,120A1)                                               E0451440
  230 WRITE (6,1110)                                                      E0451500
      IF (NEREQ(11) .EQ. 0)  WRITE (6,1120) (-VAL(I),I=1,J)               E0451600
      IF (NEREQ(11) .NE. 0) WRITE (6,1130) (-VAL(I),I=2,J)                E0451700
C                                                                         E0451800
C     ACTUAL VALUES                                                       E0451900
  250 IF (NEREQ(12) .NE. 0) J=J+1                                         E0452000
      WRITE (6,1020)                                                      E0452100
      IF(NEREQ(11) .EQ. 0 .AND. NEREQ(12) .EQ. 0) WRITE (6,1140)(KN,KN=1E0452200
     1,J)                                                                 E0452300
      IF (NEREQ(11) .NE. 0)  L=J-1                                        E0452400
      IF (NEREQ(12) .NE. 0)  L=J-1                                        E0452500
      IF (NEREQ(11) .NE. 0 .AND. NEREQ(12) .NE. 0) L=J-2                  E0452600
      IF (NEREQ(11) .NE. 0 .AND. NEREQ(12) .NE. 0)WRITE(6,1150)LOW,(KY(ME0452700
     1),M=1,L),HIGH                                                       E0452800
      IF (NEREQ(11) .EQ. 0 .AND. NEREQ(12) .NE. 0)WRITE(6,1150)(KY(M),M=E0452900
     11,L),HIGH                                                           E0453000
      IF (NEREQ(11) .NE. 0 .AND. NEREQ(12) .EQ. 0)WRITE(6,1150)LOW,(KY(ME0453100
     1),M=1,L)                                                            E0453200
      FIRST = .TRUE.                                                      E0453300
  260 NWORK=(,5+J)+1                                                      E0453400
      WRITE ( 6,1160) (ORIGN , I=1,NWORK)                                 E0453500
      IF (.NOT. FIRST)  GO TO 340                                         E0453600
      IF (IEPT(26,) GO TO 300                                             E0453700
      DO 270 N=1,2                                                        E0453800
```

Example 8-1 Contd

```
      WRITE (6,1160) ((SYM(L,M),M=1,L),L=1,J)                           F0453900
      FIRST=.FALSE.                                                     F0454000
      [illegible]                                                       F0454100
 200  WRITE (6,1200) ((SYM(L,KROWK),M=1,8),L=1,J)                       F0454200
 250  CONTINUE                                                          F0454300
      IF (.NOT. [illegible]) GO TO 300                                  F0454400
      WRITE (6,1100) ((SYM(L,1),M=1,4),ASYM(L,1),(SYM(L,1),M=1,4),L=1,J)F0454500
      DO 260 K=1,J                                                      F0454600
      KROWK=-K                                                          F0454700
 260  WRITE (6,1170)((SYM(L,KROWK),M=1,4),ASYM(L,KROWK),(SYM(L,KROWK),M=F0454800
     1 1,4),L=1,J)                                                      F0454900
 280  FIRST=.FALSE.                                                     F0455100
      GO TO 250                                                         F0455150
 300  DO 330 K=1,J                                                      F0455200
      DO 310 K=1,J                                                      F0455300
      KROWK = -K                                                        F0455400
 310  WRITE (6,1170) ((SYM(L,KROWK),M=1,8),L=1,J)                       F0455500
 320  WRITE (6,1180)  ((SYM(L,1),M=1,8),L=1,J)                          F0455600
      IF (.NOT. FIRST) GO TO 260                                        F0455700
      DO 330 K=1,J                                                      F0455800
      KROWK=-K                                                          F0455900
 330  WRITE (6,1170) ((SYM(L,KROWK),M=1,4),ASYM(L,KROWK),(SYM(L,KROWK),MF0456000
     1=1,4),L=1,J)                                                      F0456100
      WRITE (6,1190)  ((SYM(L,1),M=1,4),ASYM(L,1),(SYM(L,1),M=1,4),L=1, F0456200
     1J)                                                                F0456300
      FIRST =.FALSE.                                                    F0456400
      GO TO 300                                                         F0456500
 340  NLEVEL=J                                                          F0456600
      WRITE (6,1[illegible]) ([illegible](I),I=1,J)                     F0456700
      IF (.NOT. [illegible] .AND. NVALS .NE. 0 .AND. NLEVEL .NE. 1)     F0456800
     1 CALL [illegible] ([illegible],NPRED,IPRED,SYM,ASYM)              F0456900
      IF (.NOT.[illegible]) GO TO 400                                   F0456901
      [illegible] VALUES                                                F0456902
      ITPCNT=ITPCNT+1                                                   F0456903
      IF (ITPCNT.LT.1) GO TO 3[illegible]                               F0456904
      [illegible] =                                                     F0456905
      K=1                                                               F0456907
      IF (ITPCNT.EQ.1) GO TO 3[illegible]                               F0456908
 3[illegible] CONTINUE                                                  F0456909
```

Example 8-1 Contd

```
      DO 352 I=1,10                                                     F0456911
  352 READ (8,END=390) ITEM                                             F0456912
  354 K=K+1                                                             F0456913
      IF (ITEM.EQ.ITEM(1)) GO TO 356                                    F0456914
      GO TO 385                                                         F0456915
  356 WRITE (6,360)                                                     F0456916
  360 FORMAT (1H1,57HLISTING OF VALUES WRITTEN ON TAPE UNIT 8 UNDER OPTI F0456917
     1ON 21,//1X,9HROW COLS.)                                           F0456918
      READ (8) NR,NC                                                    F0456919
      NTIMES=1                                                          F0456922
      IF (NC.LE.130) GO TO 362                                          F0456923
      NTIMES=NC/130                                                     F0456924
      IF (MOD(NC,130).NE.0) NTIMES=NTIMES+1                             F0456925
  362 DO 385 K=1,NTIMES                                                 F0456926
      NCC=NC                                                            F0456927
      IF (NC.LE.130) GO TO 363                                          F0456928
      NCC=130                                                           F0456929
  363 DO 380 I=1,NR                                                     F0456931
      NC=NC-130                                                         F0456930
      READ(8) (UTAPE(J),J=1,NCC)                                        F0456932
      IFIRST=1                                                          F0456933
      LAST=10                                                           F0456934
      WRITE (6,367) I,IFIRST,(UTAPE(J),J=IFIRST,LAST)                   F0456935
  367 FORMAT (1H0,I3,2X,I3,10(F10.4,1X))                                F0456936
      GO TO 371                                                         F0456937
  369 WRITE (6,370) IFIRST,(UTAPE(J),J=IFIRST,LAST)                     F0456938
  370 FORMAT (6X,I3,10(F10.4,1X))                                       F0456939
  371 IF (LAST.GE.NCC) GO TO 380                                        F0456940
      IFIRST=IFIRST+10                                                  F0456941
      LAST=LAST+10                                                      F0456942
      IF (LAST.GT.NCC) LAST=NCC                                         F0456943
      GO TO 369                                                         F0456944
  380 CONTINUE                                                          F0456945
      IF (NTIMES.EQ.1) GO TO 385                                        F0456946
      WRITE (6,381)                                                     F0456947
  381 FORMAT (////1X,24HNEXT STRIP OF MAP VALUES)                       F0456948
  385 CONTINUE                                                          F0456949
  390 END FILE 8                                                        F0456950
```

Example 8-1 Contd

```
      RETURN                                                               F0456961
 1020 FORMAT (1X,62HFREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH F0457000
     C LEVEL)                                                              F0457100
 1030 FORMAT (/(1X,18A4/1X))                                               F0457200
 1040 FORMAT (/(6X,18A4))                                                  F0457300
 1060 FORMAT (//26H TOTAL MISSING DATA POINTS IS ,I5//)                    F0457400
 1080 FORMAT (1X,7HMINIMUM,12F10.2)                                        F0457900
 1090 FORMAT (1X,7HMINIMUM,8X,5HBELOW,11F10.2)                             F0458000
 1100 FORMAT (1X,7HMAXIMUM,12F10.2)                                        F0458100
 1110 FORMAT (///64H PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TF0458200
     1O EACH LEVEL)                                                        F0458300
 1120 FORMAT (//10X,12F10.2)                                               F0458400
 1130 FORMAT (//20X,11F10.2)                                               F0458500
 1140 FORMAT (1X,5HLEVEL,3X,12(I6,4X)/)                                    F0458600
 1150 FORMAT (1X,5HLEVEL,7X,11(A2,5X),A2//)                                F0458700
 1160 FORMAT (11X,12A1)                                                    F0458800
 1170 FORMAT (1H+,10X,12(1X,9A1))                                          F0458900
 1180 FORMAT (11X,12(1X,9A1))                                              F0459000
 1190 FORMAT (1X,7HSYMBOLS,1X,12(1X,9A1))                                  F0459100
 1200 FORMAT (1X,5HFREQ.,3X,12(I6,4X))                                     F0459200
      END                                                                  F0459400
                                                                            0398 CARDS
```

.4 deletes

Example 8-1 Contd

8.2 LEGENDS WITH GRID

Earlier versions of the GRID program--run on the IBM 7094 at the Harvard Computing Center--had provided for a legend package on cards and tape which would allow the inclusion of legends *similar* to those used with SYMAP. Considerable programming was necessary to make these changes compatible with the current versions of SYMAP and GRID run on the IBM 360-65.

The final solution involved the following changes to the current versions of the programs:

(i) changes in one subroutine of SYMAP so that all legends appearing on the output map are written out by row and column onto a file stored on disk; and,

(ii) changes in one subroutine in GRID so that these legends can be read in by rows and columns and shown on the map output by GRID in the same way they were shown by SYMAP. The maps must be for the same area and at the same size.

For the user the only considerations are specifying the versions of SYMAP and GRID edited to perform this task. First of all, the special version of SYMAP is used with only the legends and map packages. In the F-MAP package electives 1 and 2 should be specified as usual, elective 3 as one level, and elective 7 with some symbol *not* used in the legends for both the level and background symbolism. The legends package should be specified as usual. This separate run of SYMAP will create the necessary legends file on disk.

The edited version of the GRID program is used, for mapping; no other changes are necessary from a normal run with the GRID program.

The changes in SYMAP occur only in Subroutine OUT, shown in Example 8-2. These changes cause the legends to be transferred onto the specified disk file.

For each cell which has no symbol for level or background--and therefore does have a legend--one record will be written on a disk data set containing all the information needed for the GRID program. The format for these records is (2I5, 4A1). The first integer is the row of the legend, and the second integer is the column of the legend; the four characters are the characters for one cell, one character per overprint possible.

Changes must be made in Subroutine FLATON of the GRID program, shown in Example 8-3. In this way the legends may be read directly from disk data set unit 8.

8.3 C-LEGENDS PACKAGE

The C-LEGENDS package has been made obsolete by the C-OTOLEGENDS package and consequently is not included in the present computer program manuals. However, this package was used for the study area mapping to show legends that did not change in scale.

The C-LEGENDS package can be used to show any of the legends prepared with the C-OTOLEGENDS package; however, the legends will be incorrect if scale changes occur and

```
      SUBROUTINE OUT (IROW)
C     WRITES OUT ONE LINE OF MAP STRIP, INCLUDING OVERPRINTS IF ANY     E0279400
C     SECOND PART WRITES LEGEND ON TAPE                                 E0279500
C
      COMMON /NUMBER/ NPTS,NVALS,NINC,NBOUN,NFIG,NLEG,NCOV,NBARS,NINDL, E0279600
     1 NSLOPS                                                           E0279700
      COMMON /OUTPUT/ IOUT,IADD,MAPALL(4,131)                           E0279800
      COMMON /OPTION/ IOPT(40)                                          E0279900
      LOGICAL IOPT                                                      E0280000
      DATA IBLANK/1H /,IPLUS/1H+/                                       E0280100
C                                                                       E0280200
      IF (IROW.EQ.1) REWIND 12                                          E0280300
C
C     FILL IN LEGENDS
      IF (NLEG .NE. 0) CALL LEGD (IROW)                                 E0280400
C                                                                       E0280500
C     WRITE LINE                                                        E0280600
      IF (.NOT.IOPT(26)) WRITE (6,1000) IBLANK,(MAPALL(1,J),J=1,IOUT)   E0280700
C                                                                       E0280800
C     WRITE OVERPRINT LINES                                             E0280900
      DO 30 I=2,4                                                       E0281000
      DO 10 J=1,IOUT                                                    E0281100
      IF (MAPALL(I,J) .NE. IBLANK) GO TO 20                             E0281200
   10 CONTINUE                                                          E0281300
      GO TO 30                                                          E0281400
   20 WRITE (6,1000) IPLUS,(MAPALL(I,J),J=1,IOUT)                       E0281500
   30 CONTINUE                                                          E0281600
C                                                                       E0281700
C     WRITE LINE                                                        E0281800
                                                                        E0281900
```

Example 8-2
SYMAP Changes for Legends

```
      IF (IOPT(26)) WRITE (6,1000) IBLANK,(MAPALL(I,J),J=1,IOUT)        E0282000
C     DEFINE DOL AS INTEGER AND ASSIGN TO IT $
      INTEGER DOL
      DATA DOL/1H$/
C     FOR ALL COLUMNS OF THE ROW CHECK IF THERE IS A LEGEND
C     ASSIGNED TO THE CELL. IF NOT TRANSFER TO 45
      IF (IROW.EQ.0.OR.IROW.EQ.64) RETURN
      DO 45 J=2,124
      IF (MAPALL(1,J).EQ.DOL) GO TO 45
      WRITE (12,1001) IROW,J,(MAPALL(I,J),I=1,4)
 1001 FORMAT (2I5,4A1,10X)
   40 CONTINUE
   45 CONTINUE
   50 CONTINUE
      RETURN                                                            E0282100
 1000 FORMAT (132A1)                                                    E0282200
      END                                                               E0282300
```

Example 8-2 Contd

```
      SUBROUTINE FLATEN(ISH,ND)
      COMMON P(20000)
      COMMON/B/ NLEVEL,IFORM,NUD,IHIST,NTX,OPT(25)
      COMMON/A/ NCD,NCA,NGD,NGA,NSHEET,NCS,NCF,NCL,NUMCH,NCST,NRST
      COMMON /SYMBLS/ SYMBOL(25,4),NS(10),TITLE(60),TEXT(800)
      COMMON /OUTPUT/ MAPALL(129,4)
      INTEGER*2 X(8 ,129),INUM(129,4)
      INTEGER BLANK,SYMBOL,SYM
      LOGICAL OPT,FIN
      DIMENSION NUM(10),STCEL(4,5),IR(4)
      DATA BLANK/1H /,PLUS/1H+/,SYM1/1HI/,SYM/1HO/
      DATA NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
      DATA STCEL/16.0,6.0,15.0,8.0,18.0,9.0,14.0,11.0,4.0,17.0,1.0,12.0,
     12.0,5.0,10.0,19.0,20.0,7.0,13.0,3.0/
      DIMENSION LEGEND(4)
C
C
C     SETS THE PARAMETERS FOR EACH SHEET OF THE MAP
      OPT(16)=.TRUE.
      FIN=.FALSE.
      IEXT=0
C
      IF (OPT(16)) REWIND 8
      IF (OPT(16)) READ (8,1001) LHORRO,LA,LEGEND
 1001 FORMAT (2I5,4A1)
      IF (ND.GT.1) LHORRO=LHORRO-NCD*(ND-1)
      LHORRO=LHORRO+1
      N=NCA
      MNCP=0
      IF(NSHEET.EQ.1) GO TO 500
      N=NCF
      IF(ISH.EQ.1) GO TO 500
      N=NCS
      MNCP=NCF+NCS*(ISH-2)
      IF(ISH.EQ.NSHEET) N=NCL
  500 JFIRST=64-(NGA*N)/2
      IX=(NGD/2)+1
      JZ=NGA/2+1
```

Example 8-3
GRID Changes for Legends

```
      KK=1
      NCH=3
      IF(NUMCH.GT.1) NCH=4
      XRST=NRST-NCD*(ND-1)
      BORDER=SYM1
      IF (OPT(16)) GO TO 181
      IF(.NOT.OPT(13)) GO TO 100
      KZ=MNCP-1
      DO 200 K=1,N
      KZ=KZ+1
      NUB=NCST+(KZ/NUMCH)
      INUM(K,1)=NUB/100
      INUM(K,2)=NUB/10-INUM(K,1)*10
      INUM(K,3)=NUB-INUM(K,1)*100-INUM(K,2)*10
      IN=MOD(KZ,NUMCH)
  200 INUM(K,4)=NS(IN+1)
  100 WRITE(9,1000)(BLANK,BORDER,(SYMBOL(25,1),J=1,129),BORDER,I=1,3)
      IF(.NOT.OPT(13)) GO TO 180
      DO 201 J=1,129
      DO 201 K=1,4
  201 MAPALL(J,K)=SYMBOL(25,1)
      DO 202 KK=1,NCH
      JX=JFIRST+JZ
      DO 204 K=1,N
      II=INUM(K,KK)+1
      MAPALL(JX,KK)=NUM(II)
  204 JX=JX+NGA
  202 WRITE(9,1000) BLANK,BORDER,(MAPALL(J,KK),J=1,129),BORDER
  180 WRITE(9,1000)(BLANK,BORDER,(SYMBOL(25,1),J=1,129),BORDER,I=1,3)
      IF(FIN) RETURN
      FIN=.TRUE.
C
  181 DO 124 IROW=1,NCD
C     BLANKS OUT A ROW TO BACK GROUND SYMBOLISM
      DO 101 I=1,NGD
      DO 101 J=1,129
      IF(I.GT.4) GO TO 101
      MAPALL(J,I)=SYMBOL(25,1)
  101 X(I,J)=25.0
C     INITIALISE THE LOOP FOR EACH ROW OF CELLS
      JB=JFIRST
      ID=(IROW-1)*NCA+MNCP
```

Example 8-3 Contd

```
      DO 130 K=1,N
      ID = ID + 1
      IF(P(ID).EQ.25.0) GO TO 130
C     INSERTS SYMBOLISM KEY FOR A GRID CELL
      DO 132 I=1,NGD
      DO 132 JY=1,NGA
      J=JB+JY
      IF(OPT(12)) GO TO 131
      X(I,J) = P(ID)
      IF(OPT(8)) GO TO 132
      IF(I.NE.IX .OR. JY.NE.JZ) GO TO 132
      X(I,J)=X(I,J)+10
      IF(P(ID).GT.20.0) X(I,J)=X(I,J)-9
      GO TO 132
  131 MAPALL(J,I)=  BLANK
      IF(STCEL(I,JY).LE.P(ID)) MAPALL(J,I)=SYM
  132 CONTINUE
  130 JB=JB+NGA
C
C
      DO 124 M=1,NGD
      IEXT=IEXT+1
      IF(OPT(12)) GO TO 191
C     CONVERTS A LINE TO SYMBOLISM
      DO 103 J=1,129
      JSYM=X(M,J)
      DO 103 K=1,4
  103 MAPALL(J,K)=SYMBOL(JSYM,K)
      GO TO190
  191 KK=M
  190 IF(.NOT.OPT(13)) GO TO 220
      IF(M.NE.IX) GO TO 220
      IF(ISH.GT.1 .AND. ISH.LT.NSHEET) GO TO 220
      NUB=XRST-(IROW-1)/NUMCH
      IR(1)=NUB/100
      IR(2)=NUB/10-IR(1)*10
      IR(3)=NUB-IR(1)*100-IR(2)*10
      IN=NUMCH-MOD((IROW-1),NUMCH)
      IR(4)=NS(IN)
      JX=JFIRST-8
```

Example 8-3 Contd

```
      DO 223 J=1,2
      DO 224 I=1,NCH
      II=IR(I)+1
      MAPALL(JX,KK)=NUM(II)
  224 JX=JX+1
  223 JX=JFIRST+N*NGA+6
C     WRITES A LINE
  220 CONTINUE
      IF (.NOT.OPT(16)) GO TO 105
C     INSERTS LEGENDS IN A LINE
  108 IF (IEXT.NE.LHORRO) GO TO 105
      IF (LA.GT.130) GO TO 205
      DO 107 K=1,4
  107 MAPALL(LA-1,K)=LEGEND(K)
C
  205 READ (8,1001,END=333) LHORRO,LA,LEGEND
      IF (ND.GT.1) LHORRO=LHORRO-NCD*(ND-1)
      LHORRO=LHORRO+1
C
      GO TO 108
  333 LHORRO=0
  105 CONTINUE
      WRITE (9,1000) BLANK,BORDER,(MAPALL(J,KK),J=1,129),BORDER
      IF(OPT(12)) GO TO 124
C     WRITES OVERPRINT LINES
      DO 123 I=2,4
      DO 120 J=1,129
      IF (MAPALL(J,I) .NE. BLANK) GO TO 121
  120 CONTINUE
      GO TO 123
  121 WRITE(9,1000) PLUS ,BLANK ,(MAPALL(J,I),J=1,129),BLANK
  123 CONTINUE
  124 CONTINUE
      IF (OPT(16)) RETURN
      GO TO 100
 1000 FORMAT (132A1)
      END
```

Example 8-3 Contd

the coding requirements are not as flexible. For instance, all print characters must be specified separately: line and area legends, therefore, require as many cards as there are print locations for the legend on the map.* This section covers the necessary information for preparing a C-LEGENDS package as used for the study area.

The package is used to specify the location and content of any special wording, numbering, or other symbolism desired on the face of the map, on or within the rectangular map border. Each row of any legend is considered a separate legend, to be treated independently. Two or more legends in the same row must also be treated independently, unless they are to appear continuously, in which case they can be combined if desired.

No single legend (or combined legends) can be longer than 50 characters. If a legend longer than that is desired, it should be divided into two or more legends, each not over 50 characters in length.

On the first card of this package, punch "C-LEGENDS" in column 1-9.

On the last card punch the number "99999" in columns 1-5.

On other cards, to be inserted between the first and last, working in any convenient sequence, punch the required coordinate locations for each separate legend, using integer numbers, followed by the legend characters or symbolism. Punch the number of the row (down from the top border) in which the legend is to appear, right justified in columns 1-3.

---

* The exception is a horizontal line, or row legend. Over-printing at selected locations can be specified for a row legend; this is not possible with the point (text) or line legends of the OTOLEGENDS package.

Punch the number of the column (across from the left border) in which the legend is to start, right justified in columns 6-10. Punch the character or characters to be used in the legend starting in column 21 (and ending, at the most, in column 70). Leave a blank space wherever a blank is desired in the legend; if a legend is to end with a blank, punch the number of the column in which it ends, right justified in columns 11-15.

Special combination symbols may be produced by overprinting with up to three additional standard characters. To achieve such overprinting, include one, two, or three additional legend cards immediately following the original legend card containing the basic (first stroke) characters. On the second card, repeat the punching shown in columns 1-15 of the first card. Then punch the integer number "2" in column 20 (to designate that this card will cause the second stroke), and the desired second stroke characters in columns 21-70. Proceed in the same manner with a third card if needed, except punch the integer number "3" in column 20--and similarly with a possible fourth card.

After the cards for this package (other than the first and last) have been punched and checked, arrange them in row sequence, starting at the top of the map and working down. This may be done on a sorting machine or by hand. Where two or more legends occur in the same row, their sequence is immaterial.

An example of a C-LEGENDS package used for the study area is shown in Example 8-4.